Administrationshandbuch imperia 9

Imperia AG
Von-der-Wettern-Straße 27
D - 51149 Köln
http://www.imperia.net

Administrationshandbuch
imperia 9

Auflage: 11.01.2012 17:41 [1420]

Copyright © 2001-2012 Imperia AG Hürth/Germany

Alle Rechte vorbehalten. Dieses Handbuch darf ohne vorherige schriftliche Genehmigung der Imperia AG weder vollständig noch auszugsweise kopiert, fotokopiert, vervielfältigt, übersetzt oder in eine elektronische oder maschinenlesbare Form übertragen werden.

1. Einleitung 1
1.1. Konventionen in dieser Dokumentation 1
1.2. System-Architektur 1
1.3. Datenhaltung und Dokumentenmodell 2
1.4. Rollenkonzept 3
1.5. Rubriken 3
1.6. Workflow 3
1.7. iWE - **imperia** -WYSIWYG-Editor 4
2. Administration des Systems 5
2.1. Erstmaliges Starten und Anmelden 5
2.2. Wartungsmodi für **imperia** 5
2.2.1. Wartungsmodus ohne Superuser-Zugriff 5
2.2.2. Wartungsmodus mit Superuser-Zugriff 6
2.3. Lizenzierung 6
2.4. Zielsysteme 7
2.4.1. Neuen Freischalt-Trigger eintragen 9
2.4.2. Freischalt-Trigger bearbeiten 10
2.4.3. Freischalt-Trigger löschen 10
2.4.4. Die Darstellung beim Freischalten ändern 10
2.5. Systemdienste 10
2.5.1. Neuen Systemdienst anlegen 11
2.5.2. Systemdienst ändern 13
2.5.3. Systemdienst löschen 13
2.6. Dokumentenhaltung 14
2.6.1. Unterschiede zum Dateisystem und Einschränkungen 14
2.6.2. Konfiguration in **imperia** 14
2.6.2.1. PostgreSQL 17
2.6.2.2. ODBC_MSSQL 17
2.6.2.3. Oracle 17
2.6.3. Konfiguration der Dokumentenhaltungsschnittstelle 17
2.6.3.1. Notwendige Perl-Module 17
2.6.3.2. PostgreSQL und Linux 18
2.6.3.3. PostgreSQL und Windows 19
2.6.3.4. Oracle und Linux 19
2.6.3.5. MySQL und Windows® 20
2.6.3.6. MySQL und Linux 20
2.6.3.7. SQL-Server und Windows® 21
2.6.4. Tabellen erzeugen 21
2.6.5. Tabellen löschen 21
2.7. One-Click-Edit 21
2.7.1. One-Click-Edit und Copy-Seiten 23
2.7.2. OCE-Skript hinzufügen 23
2.7.3. **imperia** OCE-Skripte 24
2.7.3.1. oce_edit 24
2.7.3.2. oce_clone 24
2.7.3.3. oce_erase 25
2.7.3.4. oce_delete 25
2.7.3.5. oce_history 25
2.7.3.6. oce_info 25
2.7.3.7. oce_linkcheck 26
2.7.3.8. oce_reparse 26
2.7.3.9. oce_unpublish 26
2.8. Allgemeine Einstellungen 26
2.8.1. Konfiguration 27
2.8.2. Cookie-Lebensdauer 27
2.8.3. Dokumenten-History 27
2.8.4. Java-VM 27
2.8.5. Link-Checker 27
2.8.6. Superuser Authentifizierungs-Plug-In 27
2.8.7. System-Sprache 27
2.8.8. Template-Prozessor 28

2.9. Konfiguration des Systems über die Benutzeroberfläche 28
2.9.1. Konfiguration der Schnittstelle 29
2.10. Konfiguration der Link-Konsistenzprüfung 29
2.10.1. Konfigurationsbeispiel 32
2.11. Erweiterung des Schreibtischs 32
2.11.1. Schreibtischansichten einrichten 33
2.11.2. Schreibtisch-Erweiterungen in das System einbinden 34
2.11.3. Standard-Schreibtischansichten umgehen 34
2.12. Skinning der Flexmodul-Steuerelemente 35
2.12.1. FLEXMODULE-SKIN 36
2.12.2. FLEXINSTANCE-SKIN 36
2.13. Änderungsverfolgung 36
2.13.1. Änderung der Standardfarben 36
2.14. Reparsen umfangreicher Sites 37
2.14.1. Optionen 37
2.14.2. Argumente 38
2.14.3. Beispiel 38
2.15. Das interne Mailsystem 39
3. Struktur 40
3.1. Rubriken 40
3.1.1. Neue Rubrik anlegen 40
3.1.2. Rubriken bearbeiten 41
3.1.2.1. Rubriken-Metainformationen 41
3.1.2.2. Rubriken-Metainformationen bearbeiten 41
3.1.2.3. Zugriff auf die Rubriken-Metainformationen 42
3.1.2.4. Verfügbare Sprachen für mehrsprachige Dokumente festlegen 43
3.1.3. Rubrikenrechte einstellen 43
3.1.4. Rubrikenreihenfolge ändern 44
3.1.5. Rubriken löschen 45
3.2. Metadateien 45
3.2.1. Neue Metadatei anlegen 45
3.2.2. Metadatei hochladen 46
3.2.3. Metadatei bearbeiten 47
3.2.4. Metadatei löschen 47
3.3. Templates 47
3.3.1. Neues Template anlegen 48
3.3.2. Template hochladen 48
3.3.3. Template-Vorschau 49
3.3.4. Template bearbeiten 49
3.3.5. Template löschen 49
3.4. Flexmodule 50
3.4.1. Neues Flexmodul erzeugen 50
3.4.2. Flexmodule hochladen 51
3.4.3. Flexmodule bearbeiten 51
3.4.4. Flexmodule löschen 51
3.5. Slotmodule 52
3.5.1. Neues Slotmodul erzeugen 52
3.5.2. Slotmodule hochladen 53
3.5.3. Slotmodule bearbeiten 53
3.5.4. Slotmodule löschen 54
3.6. Codeinclude-Dateien 54
3.6.1. Neues Codeinclude / Verzeichnis erstellen 54
3.6.2. Codeinclude-Dateien hochladen 55
3.6.3. Codeinclude-Dateien bearbeiten 57
3.6.4. Codeinclude-Dateien / Verzeichnisse löschen 57
3.7. Dynamische Module 58
3.7.1. Laden einer externen Datei in den Editor 58
3.8. Template-Reparser 58
3.9. Assets verschieben 60
3.10. Systemdateien abgleichen 60
3.11. Dateirechte in **imperia** verwalten 60

3.12. **imperia** Onlinehilfe i9+ 61
4. iWE - **imperia** -WYSIWYG-Editor 62
4.1. Migration vom EWE-Modul zum iWE 62
4.2. Konfiguration des iWE 62
4.3. Konfigurationsoptionen 63
4.3.1. Template-Only Optionen 63
4.3.2. Freie Optionen 64
4.3.2.1. Verhalten des Editors 64
4.3.2.2. Styles 65
4.3.2.3. HTML Ausgabe 67
4.3.2.4. Benutzer-Interface 67
5. Rubriken 69
5.1. Rubriken-Eigenschaften bestimmen 69
5.1.1. Rubriken-Informationen angeben 70
5.1.2. Workflow festlegen 70
5.1.3. Metadatei und Template bestimmen 71
5.1.4. Rubriken-Zugriff 71
5.1.4.1. Zugriffsrechte zuweisen 71
5.1.4.2. Zugriffsrecht aberkennen 72
5.1.5. Asset-Rubriken 72
5.1.5.1. Asset-Rubriken in 'Freigeschaltete Dokumente bearbeiten' anzeigen 72
6. Benutzer und Rollen 73
6.1. Neue Rolle anlegen 73
6.2. Rollendaten bearbeiten 75
6.3. Löschen einer Rolle 75
6.4. Neuen Benutzer anlegen 75
6.5. Benutzerdaten bearbeiten 76
6.6. Benutzer löschen 77
6.7. LDAP und Verzeichnisdienste 77
6.7.1. Einschränkungen 78
6.7.2. Anbindung an einen Verzeichnisdienst 78
6.7.3. Verfahren zur Anmeldung an **imperia** 78
6.7.4. Notwendige Konfiguration für LDAP 79
6.7.4.1. system.conf 79
6.7.4.2. ldap.conf 79
6.7.4.2.1. Informationen für den Verzeichnisdienst 79
6.7.4.2.2. Informationen über den LDAP-Zugang 79
6.7.4.2.3. Liste der User-Container 80
6.7.4.2.4. Liste der Attribute 80
6.7.4.2.5. Passwort-Verschlüsselung 81
6.7.4.2.6. Passwort-Kodierung 81
6.7.4.2.7. LDAP-Filter für gültige **imperia** -Benutzer 81
6.7.4.2.8. Liste der Rollen-Container 81
6.7.4.2.9. Liste der Rollen-Attribute 82
6.7.4.2.10. LDAP-Filter zur Rollen-Identifikation 82
6.7.4.2.11. Gruppierungs-Schema 82
6.7.4.2.12. Superuser-Rolle 82
6.7.4.2.13. Superuser-Zugang 82
6.7.4.2.14. Benutzeranbindung 83
6.7.4.2.15. Schreibzugriff 83
6.7.4.3. **imperia** als Frontend für einen LDAP-Verzeichnisdienst 83
6.7.4.3.1. RDN-Attribut eines neuen Benutzer-Eintrags 83
6.7.4.3.2. Objektklassen eines neuen **imperia** -Benutzers 83
6.7.4.3.3. Das RDN-Attribut eines neuen Rollen-Eintrags 83
6.7.4.3.4. Objektklassen einer neuen **imperia** -Rolle 83
6.8. Beispiele für LDAP.conf 84
6.8.1. PAM 84
6.8.2. ADIR 85
6.9. Authentifizierung über HTTP 86
6.9.1. Konfiguration 86
7. Workflows 88

7.1. Workflows verwalten 89
7.1.1. Workflow-Zugriffsrechte ändern 89
7.1.2. Neuen Workflow anlegen 89
7.1.3. Workflow bearbeiten 90
7.1.4. Workflow löschen 90
7.2. Das Grid 90
7.2.1. PlugIn-Leiste 91
7.2.2. Button-Leiste 91
7.2.3. Bedienung des Grids 91
7.2.3.1. Workflow-Schritt 91
7.2.3.2. Workflow-Schritt anlegen 92
7.2.3.3. Eigenschaften eines Workflow-Schrittes einstellen 92
7.2.3.4. Workflow-Schritt im Arbeitsbereich verschieben 93
7.2.3.5. Workflow-Schritt duplizieren 93
7.2.3.6. Workflow-Schritt löschen 93
7.2.3.7. Workflow-Schritte verbinden 94
7.2.3.8. Verbindungen lösen 94
7.2.3.9. Arbeitsblatt bewegen 94
7.2.4. Zugriffsrechte für Workflow-Schritte 94
7.2.4.1. Zugriffsrechte einstellen 94
7.2.4.2. Zugriffsrechte entfernen 95
7.3. Workflow bearbeiten 95
7.4. Workflow löschen 95
7.5. Workflow-PlugIns 96
7.5.1. Asset-Aktionen 96
7.5.2. Annotate 98
7.5.3. Archiv-Bereiniger 98
7.5.4. Publizieren und Löschen 99
7.5.5. Publizieren nach Workflow 99
7.5.6. Bearbeiten 99
7.5.7. Bearbeiten + ... 101
7.5.8. Transformation 102
7.5.9. Inhaltsprüfung 102
7.5.10. HTML-Konverter 102
7.5.11. Asset-Variante 103
7.5.12. Kopie erstellen 103
7.5.13. Dokumentenauswahl 105
7.5.14. Depublizierung 108
7.5.15. Eskalation-Start i9+ 108
7.5.16. Eskalation-Ende i9+ 109
7.5.17. Existenzprüfung 109
7.5.18. Depublikationsdatum 110
7.5.19. Genehmigung 110
7.5.20. Bedingung 111
7.5.21. Genehmigung + ... 112
7.5.22. Mailversand i9+ 114
7.5.23. Metaedit 116
7.5.24. Metadaten setzten 118
7.5.25. Metadatenselektion 118
7.5.26. Mülleimer 119
7.5.27. Mehrsprachigkeit 119
7.5.28. Platzhalter 119

7.5.29. Publikationssdatum ... 120
7.5.30. Schredder ... 120
7.5.31. SmartMeta ... 121
7.5.32. Workflow einbinden ... 121
7.5.33. Asset hochladen ... 122
7.5.34. Wiedervorlagedatum ... 124
7.5.35. Lastverteilung ... 124
8. Menü-Management ... 126
8.1. Untermenü hinzufügen ... 126
8.1.1. Untermenü einem Ordner hinzufügen ... 127
8.2. Menüs bearbeiten ... 128
8.2.1. Menüpunkt zum Menü hinzufügen ... 128
8.2.2. Menüpunkt einem Untermenü hinzufügen ... 128
8.2.3. Zugriffsrechte für Menüs einstellen ... 128
8.2.4. Untermenü löschen ... 129
8.3. Menüstruktur anzeigen ... 129
9. MVC ... 130
9.1. Model ... 130
9.2. View ... 130
9.2.1. Views debuggen ... 130
9.3. Controler ... 130
9.3.1. Controller-Berechtigungen ... 130
10. Referenzen und Zielsysteme ... 132
10.1. Global Metainfo ... 132
10.2. Referenzen ... 133
10.2.1. Referenzen parsen ... 133
10.2.2. Referenzen darstellen ... 134
10.2.3. Brechen von Referenzen ... 134
10.3. Zielsysteme ... 135
10.3.1. Zielsystem wiederherstellen ... 137
10.3.2. Beispiele zum Freischalten ... 138
11. Media Asset Management (MAM) ... 140
11.1. Mit Asset-Rubriken arbeiten ... 140
11.1.1. Asset-Rubrik anlegen ... 140
11.1.2. Asset-Rubrik bearbeiten ... 142
11.1.3. Asset-Rubrik löschen ... 142
11.2. Import von Assets aus einem Ordner oder einer Zip-Datei ... 142
11.2.1. Vorarbeiten und Konfiguration ... 144
11.2.2. Import mit Hilfe des Kommandozeilentools ... 143
11.2.3. Import über die grafische Oberfläche ... 144
11.2.3.1. Vorarbeiten und Konfiguration ... 144
11.2.3.2. Die Web-Oberfläche ... 145
11.3. Konfiguration des Media Asset Managements ... 146
11.3.1. Globale Suchfilter definieren ... 146
11.3.2. Asset-Metafelder global verfügbar machen ... 146
11.3.3. Asset-Metafelder in der Listenansicht anzeigen lassen ... 147
11.3.4. Titel-Leiste des MAM-Rubrikenbaums festlegen ... 147
11.3.5. Sortierung des Asset-Rubrikenbaums festlegen ... 147
11.3.6. Template für die Erstellung von Asset-Rubriken festlegen ... 147
11.3.7. Standard-Anzahl für Mehrfachupload definieren ... 147
11.3.8. Mehrfachupload-Optionen einrichten ... 147
11.3.9. Linkanzeige der Listenansicht konfigurieren ... 148
11.3.10. Asset-Details fixieren ... 148
11.3.11. Felder verbergen ... 148
11.3.12. Datenhaltung für Binärdateien bestimmen ... 149
11.3.13. Datei-Benennung für Assets konfigurieren ... 149
11.3.14. Unabhängige Freischaltung von Assets aktivieren ... 151
11.3.15. Verfügbare Asset-Aktionen nachträglich festlegen ... 151
11.3.16. Zugriffsrechte für Asset-Rubriken verwalten ... 151

11.3.17. Listen-Ansicht konfigurieren 151
11.3.17.1. Neue Spalten hinzufügen und löschen 151
11.3.17.2. Spaltenreihenfolge ändern 153
11.4. Referenzen 153
11.5. Einbau in ein Template 153
11.6. Mehrfachupload mit Flash 154
11.6.1. Kompatibilität 154
11.6.2. Einschränkungen 154
11.6.3. Rubrikeneinstellungen 154
11.7. Darstellung der Mime-Typen 154
12. Linktool 157
12.1. Linktool 157
12.1.1. Darstellung 158
12.1.1.1. Sortierung 158
12.1.2. Prefilter 158
12.1.2.1. Metafeld 159
12.1.2.2. Besondere Felder 159
12.1.3. Filter 159
12.1.3.1. Bildfilter 160
12.1.3.2. Verzeichnisfilter 160
13. Rubriken und Dokumente umstrukturieren 161
13.1. Rubriken umstrukturieren 161
13.1.1. Rubriken mit Dokumenten umstrukturieren 162
13.1.2. Rubriken mit Assets umstrukturieren 162
13.2. Verzeichnissse und Dateien umstrukturieren 162
13.2.1. Verzeichnisse verschieben 162
13.2.2. Verzeichnisse umbenennen 163
13.3. Interne Abläufe 163
13.4. Relevante Dateien 164
14. Der Daemon Hermes 166
14.1. Starten des Hintergrund-Daemons Hermes 166
14.1.1. Starten des Daemons unter UNIX® 166
14.1.2. Starten des Daemons unter Windows® 166
14.1.3. Der Daemon als Systemdienst unter Windows® 166
14.1.3.1. Installation 167
14.1.3.2. Deinstallation 167
14.2. Kommandozeilenoptionen 167
14.2.1. Verwendung 167
14.2.2. Optionen 168
14.2.3. Log-Datei 169
14.2.4. Plug-In-Tabelle 169
14.3. Hermes-Import 170
14.3.1. Standard-Import-Konfiguration des Hermes-Daemons 170
14.3.2. Zielrubrik bestimmen 171
14.3.3. Vorhandene Dokumente beim Import mergen 171
14.3.4. Aufbau der Import-Daten 172
14.3.4.1. ASCII-Import 172
14.3.4.2. XML-Import 172
14.3.4.3. Meta-Import 172
15. Navigationserzeugung 173
15.1. Konfiguration 173
15.2. Erstellung der Navigations-Dateien 174
16. Die **imperia** -Volltextsuche 176
16.1. Einführung 176
16.2. Eingabe von Suchanfragen und Syntax für Suchbegriffe 176
16.2.1. Groß- und Kleinschreibung 176
16.2.2. Sonderzeichen und Interpunktion 176
16.2.3. Wildcards 176
16.2.4. Alternative Schreibweisen / Worttrennungen 177
16.2.5. Umlaute 177
16.2.6. Phonetisch ähnliche Alternativschreibweisen 177

16.2.7. Stopwörter ... 177
16.2.8. Phrasensuche ... 177
16.2.9. Kategorisierte Suchanfragen ... 177
16.2.10. URL-Suche ... 177
16.2.11. Wortstammsuche ... 178
16.2.12. Gruppierung ... 178
16.2.13. Nach Metafeldinhalten filtern ... 178
16.2.14. Suchwörter verknüpfen und ausschließen, logische Operatoren ... 178
16.3. Anzeige von Suchergebnissen ... 178
16.3.1. Kontextanzeige ... 179
16.3.2. Highlighting ... 179
16.3.3. Ergebnisfilter ... 179
16.4. Berechnung der Treffer-Relevanz und Boni ... 179
16.5. Einrichten der **imperia** -Volltextsuche ... 179
16.5.1. Voraussetzungen ... 180
16.5.2. Schnellstartanleitung ... 180
16.5.3. Konfiguration der Volltextsuche ... 181
16.5.3.1. Allgemeine Syntaxkonventionen ... 181
16.5.3.2. Reguläre Ausdrücke in der Konfiguration ... 182
16.5.3.3. Die Konfiguration der Indexerstellung (index.conf) ... 182
16.5.3.3.1. Basis-Tags und -Direktiven ... 183
16.5.3.3.2. Tags in Domain-Blöcken ... 187
16.5.3.3.3. Direktiven in Exclude-Blöcken ... 190
16.5.3.3.4. Direktiven in Include-Blöcken ... 190
16.5.3.3.5. Tags und Direktiven in Directory-Blöcken ... 191
16.5.3.3.6. Option-Direktiven ... 195
16.5.3.3.7. Tags und Direktiven in MetaGroup-Blöcken ... 198
16.5.3.3.8. Dynamisch benannte Meta-Gruppen ... 199
16.5.3.4. Die Konfiguration der Suche (fts.conf) ... 200
16.5.3.4.1. Basis-Tags ... 200
16.5.3.4.2. Tags in Domain-Blöcken ... 201
16.5.3.4.3. Direktiven in Option-Blöcken ... 204
16.5.3.4.4. Direktiven in parseroption-Blöcken ... 208
16.5.3.4.5. Direktiven in Output-Blöcken ... 209
16.5.3.4.6. UserSort: Sortierung der Treffer bestimmen ... 210
16.5.3.4.7. DateBonus: Treffer mit datumsbasierten Boni sortieren ... 211
16.5.3.5. Indexerstellung ... 212
16.5.3.6. Erstellung von Suchskripten ... 213
16.5.3.7. Stopwortlisten anlegen und verwalten ... 213
16.6. Vorschläge in der Volltextsuche ... 215
16.6.1. Anpassungen an der index.conf ... 216
16.6.2. Anpassungen an der fts.conf ... 216
16.6.3. Anpassungen am Template ... 217
16.6.4. Anpassungen am Ajax-Script ... 218
17. Variablen zur System-Konfiguration ... 219
17.1. Gruppe ACLs ... 219
17.1.1. Variable ENABLE_FLEX_ACL ... 219
17.2. Gruppe Anzeige ... 219
17.2.1. Variable BOTTOMTABLE ... 219
17.2.2. Variable CATEGORY_DELETION_PROMPT ... 219
17.2.3. Variable DISABLE-FAV-MENU ... 219
17.2.4. Variable DISPLAY_BASE_CLASS ... 220
17.2.5. Variable DONT_ASK_BEFORE_EXPLODE_ALL_NODES ... 220
17.2.6. Variable FAV_MENU_LINK_ONLY ... 220
17.2.7. Variable IBS_PAGE ... 220
17.2.8. Variable LATCHING ... 220
17.2.9. Variable MAIN_MENU_LINK_ONLY ... 220
17.2.10. Variable MD-WINDOW-HEIGHT ... 221
17.2.11. Variable MD-WINDOW-WIDTH ... 221
17.2.12. Variable POPUP_WINDOW_HEIGHT ... 221
17.2.13. Variable POPUP_WINDOW_WIDTH ... 221

17.2.14. Variable TOPTABLE 221
17.3. Gruppe Authentifizierung 221
17.3.1. Variable AUTH_PLUGIN 221
17.3.2. Variable AUTH_PLUGIN_BACKUP 221
17.4. Gruppe Datenhaltungs-PlugIn Datei 221
17.4.1. Variable FILE65-DOCUMENT-DIR 221
17.5. Gruppe File Transfer 222
17.5.1. Variable OPEN_3_USE_SYSREAD 222
17.5.2. Variable SSH-OPTIONS 222
17.5.3. Variable SSH-CLIENT 222
17.5.4. Variable SSH-FILED 222
17.6. Gruppe Flexmodule 222
17.6.1. Variable ALLOW_PERL_UPLOAD 222
17.7. Gruppe Grundeinstellungen 222
17.7.1. Variable CGI-BIN 222
17.7.2. Variable CGI-DIR 222
17.7.3. Variable DEMOMODE 223
17.7.4. Variable DISABLE-FLOCK 223
17.7.5. Variable DOCTREE_PLUGIN 223
17.7.6. Variable ARCHIVE_DISABLE_DELETE 223
17.7.7. Variable FollowSymLinks 223
17.7.8. Variable IMPERIA-VERSION 223
17.7.9. Variable IMPERIA_ERROR_LOG 223
17.7.10. Variable MAX_REQUEST_SIZE 223
17.7.11. Variable MIME_TYPES 223
17.7.12. Variable SECRET_FREQUENCY 223
17.7.13. Variable SECURE_UMASK 224
17.7.14. Variable DOCUMENT-ROOT 224
17.7.15. Variable OPERATING-SYSTEM 224
17.7.16. Variable SITE-DIR 224
17.8. Gruppe HTTP Cookie 224
17.8.1. Variable COOKIE_NAME 224
17.8.2. Variable COOKIE_TTL 224
17.9. Gruppe Hermes 224
17.9.1. Variable HERMES_RECONNECT 224
17.9.2. Variable MAX_AUTO_PUBLISH_DOCS 224
17.9.3. Variable HERMES_INCOMING_DELAY 225
17.10. Gruppe History 225
17.10.1. Variable DEFAULT_HISTORY_LENGTH 225
17.10.2. Variable HISTORY_HIDE_DATE 225
17.10.3. Variable HISTORY_HIDE_DATE_SU 225
17.11. Gruppe I18N 225
17.11.1. Variable DEFAULT_LOCALE 225
17.11.2. Variable DEFAULT_LANGUAGE 225
17.11.3. Variable SYSTEM-LANGUAGE 225
17.11.4. Variable SYSTEM_LOCALE 225
17.11.5. Variable USER_LOCALE 225
17.12. Gruppe Imperia Develop Session 225
17.12.1. Variable ADDR_AUTH 225
17.12.2. Variable CRYPT_ALGO 225
17.13. Gruppe Datum und Zeit 226
17.13.1. Variable CALENDAR_TIME_FORMAT 226
17.13.2. Variable CALENDAR_DISABLED 226
17.13.3. Variable DATE_COMBO_MINUTES_STEP 226
17.13.4. Variable DATE_COMBO_YEARS_LIMIT 226
17.13.5. Variable DISPLAY_TIME_ZONE 226
17.14. Gruppe Link-Checker 227
17.14.1. Variable LINK_CHECK 227
17.14.2. Variable NOPROXY 227
17.14.3. Variable PROXY_FTP 227
17.14.4. Variable PROXY_HTTP 227

17.15. Gruppe Lock Constants 227
17.15.1. Variable LOCK_EX 227
17.15.2. Variable LOCK_NB 227
17.15.3. Variable LOCK_SH 227
17.15.4. Variable LOCK_UN 227
17.16. Gruppe Media Asset Management 228
17.16.1. DISABLE_8BIT_FILENAMES 228
17.16.2. Variable MAM_CAT_ORDER_BY_NAME 228
17.16.3. Variable MAM_CATEGORY_TEMPLATE 228
17.16.4. Variable MAM_DISABLE 228
17.16.5. Variable MAM_ENABLE_PUBLISH 228
17.16.6. Variable MAM_GET_UPLOAD_FILENAME 228
17.16.7. Variable MAM_HIDDEN_FIELDS 228
17.16.8. MAM_JPEG_QUALITY 229
17.16.9. Variable MAM_LINKS_STYLE 229
17.16.10. Variable MAM_MULTIUPLOAD_OPTIONS 230
17.16.11. Variable MAM_MULTIUPLOAD_SIZE 230
17.16.12. Variable MAM_NO_EXTENDER_NORMALIZE 230
17.16.13. Variable MAM_PUBLIC_SPOOL 230
17.16.14. Variable MAM_READ_ONLY_FIELDS 230
17.16.15. Variable MAM_SAVED_FILTERS 230
17.16.16. MAM_SPOOL_CHARSET 230
17.16.17. Variable MAM_SPOOL_PATH 231
17.16.18. Variable MAM_UNIQUE 231
17.17. Gruppe Auslieferung 231
17.17.1. Variable DYNAMIC_URI_PREFIX 231
17.18. Gruppe One-Click-Edit 231
17.18.1. Variable ONECLICKEDIT 231
17.19. Gruppe PHP 231
17.19.1. Variable PHP_CLI 231
17.19.2. Variable PHP_PATH 231
17.20. Gruppe Publisher 231
17.20.1. Variable DEFAULT_VERBOSITY 231
17.20.2. Variable INFORM_ROLE_ON_PUBLISH 231
17.20.3. Variable SERIALIZER 232
17.21. Gruppe Registrierungsschlüssel 232
17.21.1. Variable REG_DEMO 232
17.21.2. Variable REG_KEY 232
17.21.3. Variable REG_NAME 232
17.22. Gruppe Reparsen 232
17.22.1. Variable REPARSE_ACTION_CHECKED 232
17.22.2. Variable REPARSE_ACTION_FIXED 232
17.22.3. Variable REPARSE_SOURCE_CHECKED 232
17.22.4. Variable REPARSE_SOURCE_FIXED 232
17.23. Gruppe Rubriken 233
17.23.1. Variable CATEGORY_TEMPLATE 233
17.23.2. Variable DEFAULT_DIRECTORY 233
17.23.3. Variable DEFAULT_FILENAME 233
17.24. Gruppe Schreibtisch 233
17.24.1. Variable DESKTOP_COLUMN_MODULES 233
17.24.2. Variable DESKTOP_CONFIG_MODULES 233
17.25. Gruppe SiteActive 233
17.25.1. Variable SAVEPAGE-PASSWORD 233
17.25.2. Variable SCHEDULE 233
17.26. Gruppe Zielsysteme 233
17.26.1. Variable live-url 233
17.26.2. Variable liveftp1-directory 233
17.26.3. Variable liveftp1-ip 234
17.26.4. Variable liveftp1-name 234
17.26.5. Variable liveftp1-pass 234
17.26.6. Variable liveftp1-site 234

17.26.7. Variable liveftp1_passive_mode 234
17.26.8. Variable transfer 234
17.27. Gruppe Template 234
17.27.1. Variable ALLOW_TEMPLATE_FORM 234
17.27.2. Variable ALLOW_TEMPLATE_FRAME 234
17.27.3. Variable ALLOW_TEMPLATE_SCRIPT 234
17.27.4. Variable MD-DEFAULT-LINKS 234
17.27.5. Variable SAFE_TEMPLATE 234
17.27.6. Variable TEMPLATE_EWE 234
17.28. Gruppe Templateprozessor 234
17.28.1. Variable TEMPLATE_CHAIN 234
17.29. Gruppe Benutzer-Management-Einstellungen 235
17.29.1. Variable DATABASE_STORAGE_PLUGIN 235
17.29.2. Variable SORT_ROLES 235
17.29.3. Variable UMLIGHT_PLUGIN 235
17.30. Gruppe Workflow 235
17.30.1. Variable CONFIRM_NEXT_STEP 235
17.30.2. Variable APPROVAL_DISABLE_JUMP 235
18. Die **imperia** Admin-Skripte 236
18.1. admin_accesscat.pl 236
18.2. admin_accessmeta.pl 236
18.3. admin_accesstemplate.pl 236
18.4. admin_accessworkflow.pl 236
18.5. admin_acl_update.pl 236
18.6. admin_addcat.pl 236
18.7. admin_adddoc.pl 237
18.8. admin_addrole.pl 237
18.9. admin_adduser.pl 237
18.10. admin_cleandesktop.pl 237
18.11. admin_currentfile.pl 237
18.12. admin_delrole.pl 237
18.13. admin_deluser.pl 237
18.14. admin_deriverole.pl 238
18.15. admin_expire.pl 238
18.16. admin_getcat.pl 238
18.17. admin_getdoc.pl 238
18.18. admin_getrole.pl 238
18.19. admin_getuser.pl 238
18.20. admin_grpaccess.pl 238
18.21. admin_grpuser.pl 238
18.22. admin_listcat.pl 238
18.23. admin_listflex.pl 239
18.24. admin_listmeta.pl 239
18.25. admin_listrole.pl 239
18.26. admin_listtemplate.pl 239
18.27. admin_listuser.pl 239
18.28. admin_listworkflow.pl 239
18.29. admin_sanitisearchive.pl 239
18.30. admin_searchdoc.pl 239
18.31. admin_searchuser.pl 239
18.32. admin_setcat.pl 240
18.33. admin_setcounter.pl 240
18.34. admin_setdoc.pl 240
18.35. admin_setrole.pl 240
18.36. admin_setuser.pl 240
18.37. admin_statdoc.pl 240
Index 241

Abbildungsverzeichnis

1.1. **imperia** Architektur 1
2.1. Login-Maske 5
2.2. Dialog Registrierungsschlüssel 7
2.3. Dialog Zielsysteme 7
2.4. Ein Zielsystem 7
2.5. Immer Freischalten 9
2.6. Metafeldtrigger 9
2.7. Benutzerbedingung 10
2.8. Dialog zum Anlegen eines Systemdienstes 11
2.9. Dialog Neuen Systemdienst anlegen 12
2.10. Dialog Dokumentenhaltung 15
2.11. Dialog OCE-Konfiguration 22
2.12. Dialog Allgemeine Einstellungen 27
2.13. Übersicht der editierbaren Konfigurationsdateien 28
2.14. Der Editor für Konfigurationsdateien 29
2.15. Die Datei `config.conf` 29
3.1. Dialog Rubriken bearbeiten 40
3.2. Rubriken-ACL-Verwaltung 44
3.3. Dialog Metadateien bearbeiten 45
3.4. Dialog Neue Metadatei anlegen 46
3.5. Dialog Metadatei hochladen 46
3.6. Dialog Templates bearbeiten 48
3.7. Dialog Neues Template anlegen 48
3.8. Dialog Template hochladen 49
3.9. Dialog Flexmodule bearbeiten 50
3.10. Dialog Neues Flexmodul erzeugen 50
3.11. Dialog Flexmodul hochladen 51
3.12. Dialog Slotmodule bearbeiten 52
3.13. Dialog Neues Slotmodul erzeugen 53
3.14. Dialog Slotmodul hochladen 53
3.15. Der Dialog Codeincludes bearbeiten 54
3.16. Der Dialog Neue Codeinclude-Datei anlegen 54
3.17. Der Dialog Codeinclude-Datei hochladen 55
3.18. Das Bearbeitungsfenster für Codeincludes 57
3.19. Dialog Template-Reparser 59
3.20. Dialog Rechtevergabe, Beispiel Metadateien 60
6.1. Dialog Rollenauswahl 75
6.2. Dialog Benutzerauswahl 76
7.1. Dialog Ändern der Workflow-Zugriffsrechte 89
7.2. Dialog Workflow anlegen 89
7.3. Das Grid 90
7.4. Dialog Workflow-Schritt Eigenschaften (Edit-PlugIn) 93
7.5. Konfiguration des PlugIns Asset-Aktionen 97
7.6. Konfiguration des PlugIns Asset-Aktionen 98
7.7. Konfiguration des Bearbeiten-PlugIns 100
7.8. Beispiel Bearbeiten + PlugIn 101
7.9. Konfiguration des PlugIns Transform 102
7.10. Konfiguration des PlugIns Asset-Variante 103
7.11. Konfiguration des PlugIns "Kopie erstellen" 104
7.12. Konfiguration des PlugIns DocSelector 106
7.13. Konfiguration des PlugIns Depublikationsdatum 110
7.14. Konfiguration des Genehmigen PlugIns 111
7.15. Eigenschaften des Bedingung-PlugIns 112
7.16. Konfiguration des PlugIns Genehmigung + ... 113
7.17. Dialog Inhaltsprüfung 113
7.18. Eigenschaften des Mailversand-PlugIns 115
7.19. Eigenschaften des Metaedit-PlugIns 117
7.20. Eigenschaften "Metadaten setzten" 118
7.21. Eigenschaften des PlugIns Metadatenselektion 119

7.22. Konfiguration des PlugIns Publikationssdatum 120
7.23. Beispiel-Workflow ohne "Workflow einbinden" 121
7.24. Workflow einbinden 122
7.25. Der abgeänderte Haupt-Workflow 122
7.26. Konfiguration des "Workflow einbinden" 122
7.27. Konfiguration des PlugIns "Asset hochladen" 123
7.28. Konfiguration des PlugIns Lastverteilung 125
11.1. Upload-Feld mit Möglichkeit zur Eingabe des Dateinamens 150
11.2. Der Dialog Neue Spalte 152
11.3. Der Dialog Anpassen 152
11.4. Dialog Asset-Referenzen 153
14.1. Beispiel Konfiguration der XYNTService.ini 167

Tabellenverzeichnis

1.1. Schlüssel-Wert-Paare eines neuen Dokuments 2
2.1. SQL-Datenbanken für die Dokumentenhaltung 14
2.2. RDBMS spezifische Treiber 18
2.3. CPAN-Module 18
7.1. 98
7.2. 98
7.3. 99
7.4. 99
7.5. 99
7.6. 101
7.7. 102
7.8. 102
7.9. 102
7.10. 103
7.11. 103
7.12. 105
7.13. 108
7.14. 108
7.15. 109
7.16. 109
7.17. 110
7.18. 110
7.19. 111
7.20. 112
7.21. 114
7.22. 116
7.23. 118
7.24. 118
7.25. 119
7.26. 119
7.27. 119
7.28. 120
7.29. 120
7.30. 121
7.31. 121
7.32. 122
7.33. 124
7.34. 124
7.35. 125
11.1. Verfügbare Felder im Media Asset Management 149
11.2. Darstellung der Mime-Typen 155
16.1. Shortcuts für logische Operatoren 178
16.2. Einstellungen der Direktive *verbosityLevel* 198
16.3. allgemein verfügbare Metafelder 198
16.4. Einstellungen für queryMode: verschiedene Suchmodi 206
16.5. Sortiermethoden 210
16.6. Zeiteinheiten für die Direktive `bonus_date_period` 211
16.7. Angaben für die Variable bonus_date_field 212
16.8. Einstellungen der Direktive *verbosityLevel* 212
16.9. Parameter für `site/bin/fts_blacklist.pl` 214
17.1. Verfügbare Felder im Media Asset Management 229

Kapitel 1. Einleitung

Dieses Kapitel gibt einen Überblick über die Funktionsweise des **imperia** CMS. Das System funktioniert aufgrund verschiedener Konzepte und Komponenten, die wir in diesem Kapitel behandeln. Hierzu gehören:

- System-Architektur
- Datenhaltung und Dokumentenmodell
- Rollenkonzept
- Workflow
- iWE

Einzelheiten zur Installation und zum Betrieb **imperia** entnehmen Sie bitte dem Installationshandbuch **Installation** [../install/install.pdf#installation] im Installationshandbuch.

1.1 Konventionen in dieser Dokumentation

Die vorliegende Dokumentation benutzt folgende Konventionen:

- Dialoge, Abschnitte, Eingabefelder, Markierungsfelder, kurz: alle Programmbestandteile werden

  `entsprechend formatiert`

  dargestellt.
- Auf besondere Zusammenhänge wird mit der Überschrift „**Achtung:**“ oder „**Warnung:**“ hingewiesen.
- Hinweise werden **fett** dargestellt.
- Code-Beispiele und Scriptaufrufe werden in

  `Festbreitenschrift`

  dargestellt.

1.2 System-Architektur

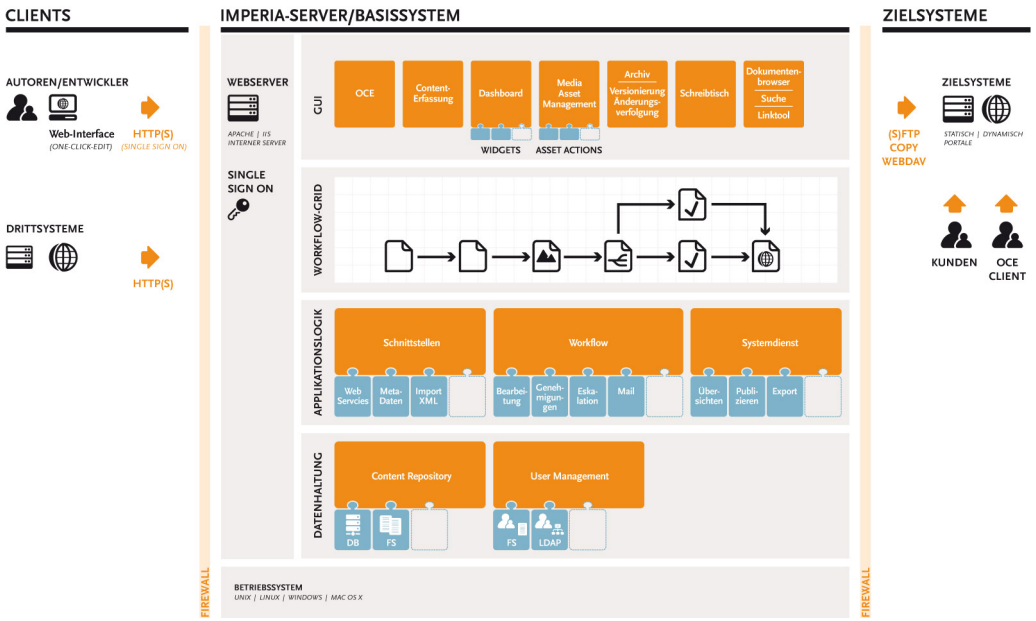


Abb. 1.1: imperia Architektur

Das **imperia** CMS besteht im Wesentlichen aus zwei Systemen: dem Basissystem (historische Synonyme: Developsystem oder Develop) und den Zielsystemen (historische Synonyme: Livesysteme oder Live).

Das Basissystem ist der Teil von **imperia** , in dem Dokumente von Autoren erzeugt und bearbeitet werden. Welche Arbeiten im Einzelnen an den Dokumenten durchgeführt werden, hängt unter anderem sehr eng mit den im Basissystem angelegten Rollen und Workflows zusammen.

Ist die Arbeit an einem Dokument erledigt, wird es, entweder manuell oder automatisch, vom Basissystem auf die Zielsysteme übertragen. Diesen Vorgang nennt man **freischalten**. Das Freischalten geschieht über Freischalt-Plug-Ins, mit deren Hilfe verschiedene Zielformate auf dem Zielsystem erzeugt werden können. Verwendet man das HTML-Freischalt-Plug-In, so kopiert **imperia** die schon auf dem Basissystem im Filesystem abgelegten und über Templates generierten Dokumente in das Filesystem des entsprechenden Zielsystems. Aus diesen freigeschalteten Dokumenten kann das Zielsystem automatisch Übersichtsseiten generieren und so dafür sorgen, dass diese immer einen aktuellen Überblick über die Themen des Webauftritts bieten. Diese Aufgabe übernimmt **SiteActive** in Zusammenarbeit mit dem Hintergrund-Daemon **Hermes** und dem Notification-Freischalt-Plug-In. Durch dieses Notification-Plug-In werden alle Metainformationen eines Dokumentes in einem Binärformat auf das Zielsystem kopiert. Der Hintergrund-Daemon **Hermes** verarbeitet dieses Notification-File und legt auf dem Zielsystem der **imperia** -Installation eine Datenhaltung an. Zusätzlich werden über das Notification-File die sogenannten Systemdienste getriggert, die die oben erwähnten **SiteActive**-Funktionen steuern.

Seit seiner Einführung 1995 hat sich das Produkt **imperia** durch intensive Zusammenarbeit mit Kunden und Endanwendern zu einem Standardprodukt für Website- und Content-Management entwickelt. Neben zahlreichen großen Installationen mit Millionen von Seitenaufrufen pro Tag hat **imperia** auch bei mittelständischen Betrieben und Startups seine Flexibilität und Leistungsfähigkeit unter Beweis gestellt. Alle **imperia** -Module weisen folgende Gemeinsamkeiten auf:

- Sie sind plattformunabhängig.
- Es wird keine zusätzliche Client-Software benötigt; die Bedienung erfolgt ausschließlich im Web-Browser.
- Alle **imperia** -Module werden im Source-Code ausgeliefert.
- Sie sind skalierbar und modular und somit leicht erweiterbar.
- Die Nutzung einzelner Module wird durch Benutzer- und Rollenrechte geregelt.
- Die Datenhaltung erfolgt im Dateisystem oder in einer Datenbank.

1.3 Datenhaltung und Dokumentenmodell

imperia bietet verschiedene Möglichkeiten der Datenhaltung. Einmal kann man die Daten im Dateisystem speichern. Diese Möglichkeit ist bei kleiner und mittlerer Projektgröße sehr performant. Eine weitere Möglichkeit der Dokumentenhaltung ist eine SQL-Datenbank. **imperia** unterstützt eine Reihe von kommerziellen und freien SQL-Datenbanken.

Nach der Installation speichert **imperia** die Dokumente im Dateisystem. Wenn eine andere Möglichkeit der Datenhaltung verwendet werden soll, sollte dies eingestellt werden, bevor Dokumente erzeugt werden. Aber auch ein nachträgliches Umschalten der Datenhaltung ist möglich - dazu stellt **imperia** entsprechende Admin-Skripte zur Verfügung.

Ein **imperia** -Dokument ist eine Sammlung von Informationen. Neben dem eigentlichen Dokumenteninhalt gehören dazu auch Angaben, die das Dokument beschreiben, beispielsweise der Verfasser und das verwendete Template. **imperia** speichert sämtliche Informationen eines Dokuments in so genannten Metafeldern. Diese sind als Schlüssel-Wert-Paare aufgebaut, wobei der Schlüssel den Namen des betreffenden Feldes festlegt und der Wert dessen Inhalt. Jeder Schlüssel steht für eine Meta-Variable, jeder Wert für den Inhalt der Meta-Variable.

Beim Erzeugen eines Dokuments werden bereits einige Metafelder angelegt und gefüllt, so dass ein gerade erzeugtes Dokument unter anderem die folgenden Schlüssel-Wert-Paare besitzt:

Schlüssel	Beispielwert
`directory`	`/news/aktuell`
`template`	`myTemplate`
`metafile`	`mymeta`

Tabelle 1.1. Schlüssel-Wert-Paare eines neuen Dokuments

Diese Schlüssel und Werte erbt das Dokument von den Rubriken-Eigenschaften der Rubrik, die der Benutzer beim Erzeugen des neuen Dokuments ausgewählt hat. Jedes der Schlüssel-Wert-Paare kann jedoch an jeder Stelle des Workflows verändert werden, zum Beispiel durch ein Workflow-Plug-In (Metasetter) oder durch einen Benutzer in einem Edit-Schritt.

1.4 Rollenkonzept

Ein wesentlicher Bestandteil des Produktionssystems sind Rollen. Rollen verleihen einem Benutzer Rechte auf **imperia** -Module, zum Beispiel auf das Media Asset Management (MAM) oder das Archiv, und auf Programmfunktionen, wie zum Beispiel **Neues Dokument erzeugen** oder **Dokumente freischalten**.

Ferner verleihen Rollen dem Benutzer Rechte auf Rubriken, die bestimmte Bereiche des Projekts darstellen. Er kann dann nur in diesen Projektbereichen die Funktionen ausüben, die ihm seine Rollen verleihen.

Rollen werden häufig so angelegt, dass jede Rolle bestimmte Funktionen oder Zugriff auf bestimmte Rubriken verleiht. Jeder Benutzer kann gleichzeitig mehreren Rollen angehören, so dass er die Summe der Rechte aller angenommenen Rollen erhält. Die Kombination mehrerer Rollen gibt einem Benutzer also die Rechte, die er benötigt, um die für ihn vorgesehenen Funktionen in den für ihn vorgesehenen Projektbereichen auszuführen.

Weitere Informationen finden Sie in Kapitel 6 **Benutzer und Rollen**.

1.5 Rubriken

Die unterschiedlichen Themengebiete eines Projektes lassen sich in **imperia** durch Rubriken repräsentieren. Diese lassen sich jeweils in mehrere Unterrubriken gliedern. Dabei sind die Unterrubriken bei der Erstellung mit den Einstellungen der übergeordneten Rubriken vorbelegt.

Über die **imperia** -Rechteverwaltung legen Sie fest, wer in den einzelnen Rubriken lesen und schreiben bzw. Dokumente erstellen darf. Damit ein Benutzer in einer Unterrubrik Dokumente erzeugen kann, muss er in allen Rubriken, die im Ast über dieser Rubrik liegen, mindestens Leserechte haben.

Jedes Dokument, das in einer Rubrik erzeugt wird, erhält davon Eigenschaften, von denen einige in Meta-Variablen des Dokuments gespeichert werden:

- Verzeichnis (Meta-Variable `directory`)
- Template (Meta-Variable `template`)
- Metadatei (Meta-Variable `metafile`)
- Workflow-Einstiegspunkt

Weitere Informationen finden Sie in Kapitel 5 **Rubriken**

1.6 Workflow

Häufig haben sich in einem Unternehmen bereits eine Reihe von Arbeitschritten etabliert, mit denen Inhalte ins Internet, Intranet und/oder Extranet gestellt werden. Hierzu gehören zum Beispiel Arbeitsschritte wie *beschreibende Informationen zu einer Seite eingeben*, *Inhalte in eine Seite eingeben* und *neue Seite überprüfen und freigeben oder ablehnen*.

In **imperia** wird eine Reihe von Arbeitsschritten als Workflow bezeichnet. **imperia** -Workflows sind völlig frei definierbar, man kann fast jede denkbare Reihe von Arbeitsschritten abbilden. Jedes Dokument muss seinen Workflow durchlaufen, bevor es fertig ist und freigeschaltet werden kann.

Jeder Schritt eines Workflows wird durch ein Workflow-Plug-In abgebildet. Im Lieferumfang sind eine Reihe von Workflow-Plug-Ins enthalten, die eine Vielzahl von Workflow-Funktionen zur Verfügung stellen. Durch das Aneinanderreihen und Verbinden der einzelnen Plug-Ins wird der Workflow gebildet. Als Tool zum Erzeugen und Verändern von Workflows steht eine grafische Oberfläche zur Verfügung - das so genannte Grid.

Man unterteilt Workflow-Plug-Ins in interaktive und nicht-interaktive Plug-Ins. Interaktive Plug-Ins erwarten an dieser Stelle des Workflows die Eingabe eines Benutzers. Ein Beispiel hierfür ist das Bearbeiten-Plug-In. Dieses Plug-In stellt das Template in der Edit-Ansicht dar und erwartet, dass ein Benutzer Inhalt eingibt und seine Änderungen speichert.

Ein nicht-interaktives Plug-In ist beispielsweise das Metasetter-Plug-In, das eine Meta-Variable eines Dokuments auf einen eingestellten Wert setzt. Hierdurch können zum Beispiel auch Steuerungsvariablen für den Workflow definiert werden.

Neben Plug-Ins, die Meta-Variablen eines Dokuments manipulieren, gibt es auch Plug-Ins, die den Wert von Meta-Variablen überprüfen. Ein Beispiel hierfür ist das IfElse-Plug-In, das abhängig von dem Wert einer Meta-Variablen den weiteren Weg des Dokuments durch den Workflow ändern kann.

Man hat also sowohl die Möglichkeit, Meta-Variablen in einem Dokument zu setzen als auch die Möglichkeit, den Wert einer Meta-Variablen zu überprüfen und daraufhin den weiteren Workflow eines Dokuments zu bestimmen. Diese beiden Möglichkeiten machen den Workflow in **imperia** zu einem sehr mächtigen und komfortablen Werkzeug, wenn es darum geht, den bereits im Unternehmen vorhandenen Workflow in **imperia** nachzubilden.

Weitere Informationen finden Sie in Kapitel 7 **Workflows**

1.7 iWE - imperia -WYSIWYG-Editor

Der iWE (**imperia** -WYSIWYG-Editor) ist der WYSIWYG Editor in **imperia** . Er basiert auf der zweiten Version des FCK-Editors. Durch entsprechende Anpassungen wurde er auf die Anforderungen des **imperia** CMS angepasst.

Weitere Informationen finden Sie in Kapitel 4 **iWE - imperia -WYSIWYG-Editor**

Kapitel 2. Administration des Systems

Alle für die Systemadministration relevanten Menüs und Dialoge werden über den Ordner **System** geöffnet.

2.1 Erstmaliges Starten und Anmelden

Um **imperia** zum ersten Mal zu starten, geben Sie in der Adressleiste Ihres Browsers folgenden Link ein:

```
http://<Ihr Produktions-Server>/imp/
```

Es wird die Login-Maske geöffnet:

Abb. 2.1: Login-Maske

Zum erstmaligen Anmelden an **imperia** wurde bei der Installation der Benutzer **superuser** angelegt. Als Passwort wurde für diesen Benutzer ebenfalls **superuser** eingerichtet. Der Superuser hat Zugriff auf alle Funktionen und Menüs, konfiguriert und pflegt das System und legt Benutzer und Rollen an.

Achtung:

Der Superuser hat alle Berechtigungen im System. Arbeiten Sie unter diesem Benutzer nur, wenn Sie das entsprechende Wissen über die einzelnen Menüpunkte haben!

Starten Sie **imperia** und tragen Sie in das Feld **Name** der Login-Maske **superuser** ein. In das Feld **Password** tragen Sie ebenfalls **superuser** ein. Klicken Sie anschließend auf den Button **Okay**. Daraufhin wird das Menü geöffnet.

Vervollständigen Sie nun die Konfiguration von **imperia** . Führen Sie dazu folgende Schritte durch:

1. Ändern Sie das voreingestellte Passwort für den Superuser.
2. Installieren Sie ein Zielsystem, falls dies noch nicht geschehen ist. Lesen Sie hierzu ???.
3. Tragen Sie die Lizenzdaten ein, damit alle Funktionen freigeschaltet werden. Lesen Sie hierzu Abschnitt 2.3 **Lizenzierung** auf Seite 6.
4. Geben Sie die Daten des Zielsystemes ein. Lesen Sie hierzu Abschnitt 2.4 **Zielsysteme** auf Seite 7.

2.2 Wartungsmodi für imperia

Für einige administrative Aufgaben ist es sinnvoll, vor der Durchführung dieser Wartungsarbeiten das Produktionssystem in einen Wartungsmodus zu versetzen. Im Wartungsmodus können die Benutzer nicht auf das System zugreifen, ohne dass der Webserver gestoppt werden muss.

In **imperia** stehen zwei unterschiedliche Wartungsmodi zu Verfügung, die den Zugriff auf das System unterschiedlich stark einschränken.

2.2.1 Wartungsmodus ohne Superuser-Zugriff

Sobald dieser Wartungsmodus aktiviert wird, werden die Klicks aller Benutzer (auch des Superusers) in die Login-Maske umgelenkt, wo sie sich erst wieder anmelden können, wenn der Wartungsmodus wieder abgeschaltet wurde.

Um das Produktionssystem in diesen Wartungsmodus zu versetzen, setzen Sie folgende Variable in der Datei `site/config/system.conf`:

```
"AUTH_PLUGIN" = "MaintenanceNoLogin"
```

Hinweis:

Sie müssen diese Variable von Hand in der `site/config/system.conf` *nachtragen, da sie bei der Installation nicht automatisch dort angelegt wird.*

Um den Wartungsmodus wieder zu verlassen, müssen Sie den Wert der `system.conf`-Variablen wieder auf den Standard-Wert `Password` setzen oder die entsprechende Zeile auskommentieren (mit vorangestellter Raute #) bzw. löschen.

Dieser Modus bietet sich beispielsweise an, wenn Sie unter einer SQL-Datenhaltung ein Backup Ihrer Datenbank erstellen wollen und verhindern möchten, dass während des Backup-Prozesses durchgeführte Änderungen am Content verloren gehen.

2.2.2 Wartungsmodus mit Superuser-Zugriff

Während der erste Wartungsmodus alle Log-In-Versuche unterbindet, kann sich in diesem Modus der Superuser am System anmelden. Das kann bei bestimmten Wartungsarbeiten erforderlich sein.

Hinweis:

Es kann sich in diesem Modus nur der Superuser anmleden, nicht die inhaber der Rolle Superuser.

Die Aktivierung können Sie entweder im Dialog *Allgemeine Einstellungen* vornehmen (siehe Abschnitt 2.8.6 **Superuser Authentifizierungs-Plug-In** auf Seite 27) oder indem Sie die Datei `system.conf` editieren:

```
"AUTH_PLUGIN" = "Superuser"
```

Um das System wieder in den Normalbetrieb zu versetzen, verfahren Sie wie beim Wartungsmodus ohne Superuser-Zugriff bzw. deaktivieren Sie die entsprechende Option im Dialog *Allgemeine Einstellungen*.

2.3 Lizenzierung

Die Lizenzierung schaltet die erworbenen Funktionen in **imperia** frei. Mit der Software erhalten Sie einen Registrierungsschlüssel und den Namen des Lizenznehmers. Beide werden in den Dialog Registrierungsschlüssel eingetragen. Achten Sie hierbei auf Groß- und Kleinschreibung oder Leerstellen.

Falls in Zukunft weitere Funktionen oder Module erworben werden, müssen die alten Daten durch die neuen Daten ersetzt werden.

Zur Registrierung von **imperia** gehören:

- die Datei `regkey.bin`
- der Name des Lizenznehmers
- ein 24-stelliger Registrierungsschlüssel

Kopieren Sie zunächst die Datei `regkey.bin` in das Verzeichnis `site/config` auf dem Produktionssystem. Die Datei finden Sie auf Ihrer CD. Starten Sie anschließend **imperia** und melden Sie sich als Benutzer mit entsprechenden Rechten an, im Zweifel als Superuser. Öffnen Sie den Dialog **Registrierungsschlüssel** über den Link **Registrierungsschlüssel** des Ordners **System**. Sie gelangen daraufhin direkt in folgenden Eingabe-Dialog:

Abb. 2.2: Dialog Registrierungsschlüssel

Tragen Sie den Lizenznehmer und den Registrierungsschlüssel in die gleichlautenden Felder des Dialogs ein. Achten Sie unbedingt auf Groß- und Kleinschreibung und auf die exakte Übernahme der Schreibweise des Lizenznehmers. Klicken Sie anschließend auf den Button **Okay**. Der Button **Abbrechen** bricht die Lizenzierung ab, ohne Eingaben zu speichern.

2.4 Zielsysteme

Um mit **imperia** Dokumente freischalten zu können, müssen die Daten mindestens eines Zielsystems bekannt sein. Diese Daten werden in den Dialog **Zielsystem** eingetragen. Der Dialog wird über den Menüpunkt **Zielsystemw** des Ordners **System** geöffnet.

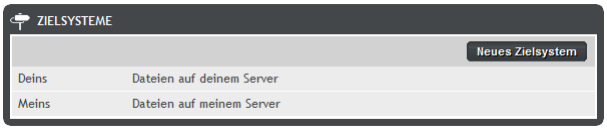


Abb. 2.3: Dialog Zielsysteme

Der Dialog zeigt die bestehenden Zielsysteme an. Mit einem Klick auf die Zeile kann das entsprechende Zielsystem bearbeitet werden..

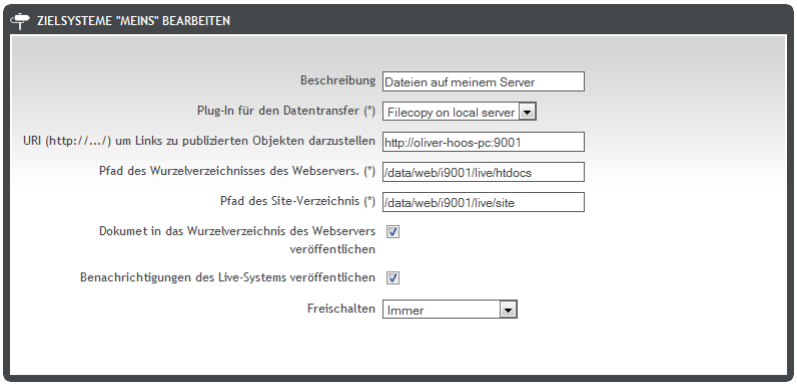


Abb. 2.4: Ein Zielsystem

Die ID des Zielsystems darf Groß- und Kleinbuchstaben enthalten, wobei IDs nur mit Großbuchstaben für interne Zwecke reserviert sind und somit nicht verwendet werden können.

Beschreibung des Zielsystems

An dieser Stelle können Sie eine Beschreibung für das Zielssystem angeben.

Plug-In für den Datentransfer

Über die Drop-Down-Box wird das Protokoll eingestellt, mit dem fertige Dokumente auf das Zielsystem übertragen werden sollen. Folgende Protokolle stehen zur Auswahl:

- Filecopy on local server
- FTP-Transfer
- SFTP-Transfer
- HTTP-Transfer
- WebDAV (mit und ohne SSL)

Zu jedem Protokoll werden das zu verwendende Protokoll, mit dem Dateien auf das Zielsystem übertragen werden (z.B. http, https, ftp), das Document-Root und das Site-Verzeichnis des Zielsystems in die entsprechenden Felder eingegeben. Alle Felder müssen ausgefüllt werden. Folgende Felder sind möglich:

Document-Root

Hier muss das Standard-Verzeichnis für HTML-Dateien eingetragen werden. Oft ist dies `/var/htdocs` oder `c:/inetpub/wwwroot`.

SITE-Verzeichnis

Beim Site-Verzeichnis handelt es sich um ein Standard-Verzeichnis für **imperia** ; es stellt die Basis für alle Module in **imperia** dar. Beispiel: `/pfad/zur/imperia/installation/site`

Die Verzeichnisse, die Sie unter Document-Root und Site-Verzeichnis eintragen, müssen bereits existieren. Achten Sie darauf, Verzeichnisangaben grundsätzlich ohne schließenden Schrägstrich (Slash: /) zu tätigen. Auch unter Windows® müssen Verzeichnisse mit Hilfe des Schrägstrichs und nicht mit Backslashes (\ ) angegeben werden.

Weitere in diesem Dialog mögliche Felder sind:

URI (http://...)

Tragen Sie hier den URI für Ihr Zielsystem ein (z.B.: http://www.ihredomain.de).

IP-Adresse

Tragen Sie hier die IP-Adresse des Zielsystems ein, wenn Dokumente mit FTP oder HTTP freigeschaltet werden. Optional können Sie hier auch den Benutzernamen ergänzen, beispielsweise bei der Verwendung von SecureCopy (wwwrun@meinserver.de).

Login

Geben Sie hier die Anmeldekennung an, falls Ihr Zielsystem dies erfordert.

Passwort

Tragen Sie hier das Passwort ein, falls das Zielsystem dies erfordert.

Passiver FTP-Modus

Wird diese Checkbox markiert, verwendet **imperia** den passiven FTP-Modus, um Dokumente per FTP auf das Zielsysteme zu übertragen.

Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Angaben zu speichern. Die aktuellen Parameter werden anschließend zusammengefasst dargestellt. Der Button **Abbruch** bricht die Aktion ab und Sie gelangen zurück in das Menü, ohne dass Eingaben gespeichert werden.

Freischalten

Über das Feld Freischalten können Sie Einschränkungen für das Freischalten von Daten festlegen. Nur Dokumente die dieses Kriterium erfüllen, werden auf dieses Zielsystem freigeschaltet.

Nach der Installation ist **imperia** so eingestellt, dass freigeschaltete Dokumente und die dazugehörigen Medien immer auf das Zielsysteme freigeschaltet werden. Im vorliegenden Abschnitt beschäftigten wir uns mit der Einrichtung sogenannter Freischalt-Trigger, die weitere Ziele für freigeschaltete Dokumente definieren und anhand derer **imperia** das Zielsystem bestimmt, auf das ein freigeschaltetes Dokumente übertragen werden soll.

Ein Freischalt-Trigger besteht aus:

- der Beschreibung des Ereignisses (Triggers), das die Übertragung der Dateien auf einen Server auslöst.
- dem Zielformat der freigeschalteten Dateien.
- dem Namen und dem Pfad der Zieldatei.

- der Angabe des Servers, auf den die Datei übertragen werden soll. Dies kann das Zielsysteme, das Produktionssystem oder ein alternatives Zielsystem sein.

Weitere Informationen zu den Bestandteilen eines Freischalt-Triggers finden Sie im folgenden Abschnitt:

2.4.1 Neuen Freischalt-Trigger eintragen

Der Konfigurationsdialog für Freischalt-Trigger wird über den Link **Zielsysteme** im Menü **System** geöffnet.

Im Abschnitt **Trigger** wird festgelegt, wodurch der Freischaltvorgang auf dieses Zielsystem ausgelöst werden soll. Abhängig von der Auswahl der Drop-Down-Box werden unterschiedliche Eingabefelder in diesem Abschnitt des Dialogs angezeigt. Folgende Möglichkeiten stehen zur Verfügung:

Immer

Der Freischalt-Trigger wird immer ausgeführt, wenn ein Dokument freigeschaltet wird.

Abb. 2.5: Immer Freischalten

Metafeld

Der Freischalt-Trigger wird ausgeführt, wenn ein bestimmtes Metafeld einen definierten Wert hat. Das betreffende Metafeld und der entsprechende Wert werden in die Eingabefelder **Feldname** und **Wert** eingetragen. Die Werte müssen als reguläre Ausdrücke eingegeben werden und werden mit Hilfe von Pattern-Matching überprüft. Ein genauer Match wird beispielsweise durch Verankerung erreicht: `^dies muss genau matchen$`.

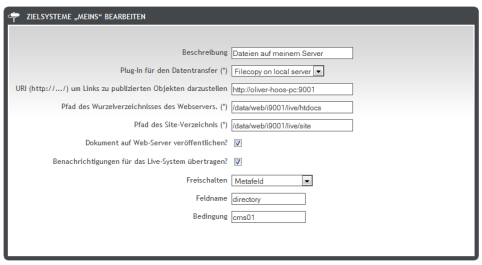


Abb. 2.6: Metafeldtrigger

Benutzerbedingung

Der Freischalt-Trigger wird ausgeführt, wenn das Profil des freischaltenden Benutzers ein Feld mit einem bestimmten Wert enthält. Das betreffende Feld des Benutzerprofils und der entsprechende Wert werden in die Eingabefelder **Feldname** und **Wert** eingetragen. Die Werte müssen als reguläre Ausdrücke eingegeben werden und werden mit Hilfe von Pattern-Matching überprüft. Ein genauer Match wird durch Verankerung erreicht: `^dies muss genau matchen$`. Ein klassische Beispiel hierbei ist die Verwendung des Feldnamens `username`. Andere Möglichkeiten bieten die Felder Email-Adresse oder Benutzer-ID.

Die Eingaben werden über den Button **Okay** am unteren Rand gespeichert. Der Button **Schließen** schließt den Dialog, ohne Eingaben zu speichern.

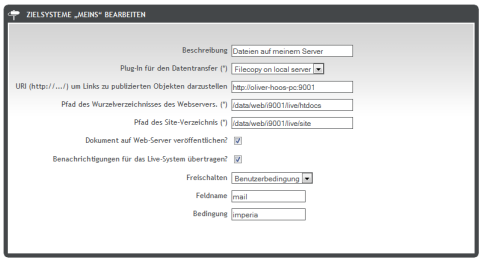


Abb. 2.7: Benutzerbedingung

2.4.2 Freischalt-Trigger bearbeiten

Um einen bestehenden Freischalt-Trigger zu ändern, wird in der Liste der eingerichteten Freischalt-Trigger der Button **Bearbeiten** in der entsprechenden Zeile ausgewählt.

1. Öffnen Sie den Konfigurationsdialog für Freischalt-Trigger über den Menüpunkt **Zusätzliche Zielsysteme** im Ordner **System**.

   Es wird eine Liste mit allen eingerichteten Freischalt-Triggern angezeigt.

2. Klicken Sie in der entsprechenden Zeile der Liste auf den Button **Bearbeiten**.

   Es wird ein weiteres Fenster geöffnet, in dem die Einstellungen des Triggers geändert werden können.

3. Ändern Sie die Einträge in den einzelnen Abschnitten Ihren Wünschen entsprechend.
4. Der Button **Okay** speichert die Änderungen. Der Button **Schließen** schließt das Fenster, ohne Änderungen zu speichern.

Die Änderung eines Freischalt-Triggers ist damit abgeschlossen.

2.4.3 Freischalt-Trigger löschen

Um einen bestehenden Freischalt-Trigger zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie den Konfigurationsdialog für Freischalt-Trigger über den Menüpunkt **Zusätzliche Zielsysteme** im Ordner **System**.

   Es wird ein weiteres Fenster geöffnet mit der Liste der bereits eingerichteten Freischalt-Trigger.

2. Klicken Sie neben dem Freischalt-Trigger, den Sie löschen möchten, auf den Button **Löschen**.

   Der ausgewählte Freischalt-Trigger wird gelöscht und die Liste wird aktualisiert.

2.4.4 Die Darstellung beim Freischalten ändern

Insbesondere bei der Verwendung zahlreicher Freischalt-Trigger kann die Darstellung der Abfrage der einzelnen Trigger sehr umfangreich werden. Mit der `system.conf`-Variable

```
DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGERS
```

kann die Darstellung angepasst werden. Informationen hierzu finden Sie im Anhang in ???.

2.5 Systemdienste

imperia startet Systemdienste zu bestimmten Zeitpunkten automatisch, um SiteActives zur Erzeugung von Übersichtsseiten oder Datenbank-Exportaufträge auszuführen. Voraussetzung für die Nutzung von Systemdiensten ist ein laufender Hermes-Daemon auf dem Entwicklungssystem und allen Zielsystemen, auf denen Sie den Systemdienst nutzen wollen. Lesen Sie hierzu auch den Abschnitt Kapitel 14 **Der Daemon Hermes**.

Die Konfiguration der auf Ihrem **imperia** -System laufenden Dienste nehmen Sie im Dialog **Menü**->**System**->**Systemdienste** vor. **imperia** speichert die Systemdienstkonfiguration in der Datei `site/config/imperia.schedule` Ihres Entwicklungsystems. Diese Datei muss auch auf allen Zielsystemen vorliegen, auf denen die Systemdienste laufen sollen. Zur Übertragung dieser Konfigurationsdatei nutzen Sie die Funktion **Menü**->**System**->**Systemdateien abgleichen**.

Den Zeitpunkt für die Ausführung können Sie entweder durch zeitlich- oder ereignisgesteuert definieren. Für zeitlich gesteuerte Systemdienste sind folgende Optionen für die Ausführung verfügbar:

- in bestimmten Intervallen (stündlich, täglich, nach Ablauf bestimmter Minuten- oder Stundenfristen)
- zu bestimmten Uhrzeiten
- an bestimmten Wochentagen
- monatlich

Ereignisgesteuerte Sysytemdienste richten Sie ein, indem Sie eines oder mehrere Verzeichnisse angeben, die der Hermes-Daemon überwachen soll. Wenn Sie in einem überwachten Verzeichnis ein neues Dokument hinzufügen oder ein bestehendes löschen oder verändern, führt der Hermes-Daemon den entsprechenden Systemdienst aus.

Beide Varianten sind voneinander unabhängig. Sie können also beispielsweise ein SiteActive sowohl durch einen zeitlich gesteuerten Systemdienst als auch bei Änderungen in einem bestimmten Verzeichnis ausführen lassen. Achten Sie bei der Kombination von ereignis- und zeitlich gesteuerten Systemdiensten darauf, dass der Hermes-Daemon das ausgeführte SiteActive nicht zu selten oder zu häufig ausführt.

Wenn Sie für ein SiteActive beispielsweise einen ereignisgesteuerten und einen zeitlich gesteuerten Systemdienst mit einem langen Intervall (z.B. monatlich) definiert haben, geschieht die Ausführung fast ausschließlich durch Änderungen im überwachten Verzeichnis. Ohne den ereignisgesteuerten Systemdienst startet nur der zeitgesteuerte Dienst das SiteActive. Bei großen Zeitintervallen besteht dann beispielsweise die Gefahr, dass gelöschte Artikel vorübergehend zu toten Links führen.

Treten hingegen in einem überwachten Verzeichnis häufig Änderungen ein und ist zusätzlich ein Systemdienst mit einem kurzen Zeitintervall von beispielsweise fünf Minuten definiert, kann es passieren, dass der Hermes-Daemon das betreffende SiteActive mehrfach innerhalb eines Durchlaufs ausführt.

Der Hermes-Daemon entnimmt der Konfiguration des Systemdienstes, welches Template für die Ausführung des SiteActives dient und wo die daraus generierte Datei unter welchem Namen abzuspeichern ist. Diese Parameter übergibt er einem CGI-Skript, dass Sie zum Testen des SiteActive-Codes auch manuell aufrufen können. Lesen Sie hierzu den Abschnitt "**SiteActives manuell aufrufen**" im Kapitel "**SiteActive**" des Programmierhandbuchs.

Achtung:

Die Ausführung eines Systemdienstes in kurzen Intervallen oder mehrerer Systemdienste gleichzeitig kann zu erheblicher Systemlast auf dem Server führen.

2.5.1 Neuen Systemdienst anlegen

Rufen Sie den Dialog für die Systemdienstkonfiguration auf: **Menü**->**System**->**Systemdienste**.

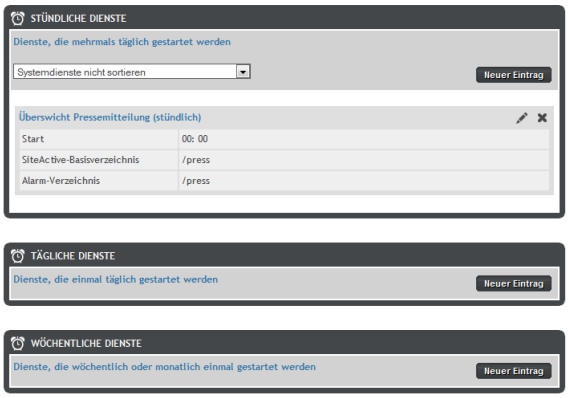


Abb. 2.8: Dialog zum Anlegen eines Systemdienstes

Klicken Sie in einem der Abschnitte auf den Link **Neuer Eintrag**. Es erscheint folgender Dialog:

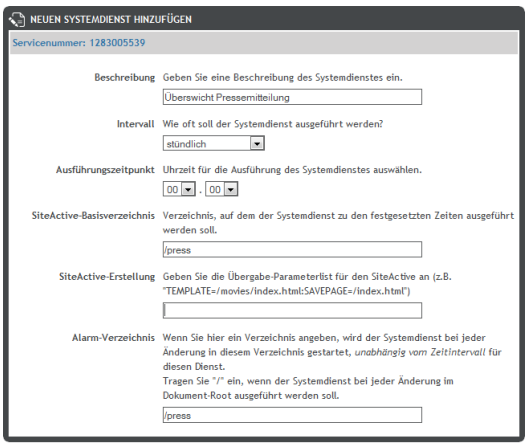

NEUEN SYSTEMDIENST HINZUFÜGEN
Servicenummer: 1283005539
Beschreibung
Geben Sie eine Beschreibung des Systemdienstes ein.
Übersicht Pressemitteilung
Intervall
Wie oft soll der Systemdienst ausgeführt werden?
stündlich
Ausführungszeitpunkt
Uhrzeit für die Ausführung des Systemdienstes auswählen.
00 . 00
SiteActive-Basisverzeichnis
Verzeichnis, auf dem der Systemdienst zu den festgesetzten Zeiten ausgeführt werden soll.
/press
SiteActive-Erstellung
Geben Sie die Übergabe-Parameterlist für den SiteActive an (z.B. "TEMPLATE=/movies/index.html:SAVEPAGE=/index.html")
Alarm-Verzeichnis
Wenn Sie hier ein Verzeichnis angeben, wird der Systemdienst bei jeder Änderung in diesem Verzeichnis gestartet, unabhängig vom Zeitintervall für diesen Dienst.
Tragen Sie "/" ein, wenn der Systemdienst bei jeder Änderung im Dokument-Root ausgeführt werden soll.
/press

Abb. 2.9: Dialog Neuen Systemdienst anlegen

Die Zuordnung eines Dienstes zu einem Abschnitt geschieht erst, wenn Sie einen Zeitpunkt für die Ausführung des Dienstes angeben. Wenn Sie also den Eingabedialog für einen neuen Systemdienst über den Abschnitt **Wöchentliche Dienste** öffnen, den Dienst jedoch viertelstündlich ausführen lassen, erscheint der Diensteintrag nach dem Speichern im Abschnitt **Tägliche Dienste**.

Jeder Systemdienst erhält automatisch eine Dienstnummer, die Sie im oberen Teil des Dialogs sehen. Das Feld **Beschreibung** dient für eine kurze Beschreibung der Funktion des Dienstes, die in der Liste der Systemdienste als Dienstüberschrift erscheint. Mit der Drop-Down-Box **Intervall** bestimmen Sie bei zeitlich gesteuerten Systemdiensten das Ausführungsintervall. Die beiden Drop-Down-Boxen unter der Überschrift **Ausführungszeitpunkt** dienen zum Einstellen einer bestimmten Uhrzeit. Mit der linken Drop-Down-Box stellen Sie die Stunden und mit der rechten die Minuten im Viertelstundentakt ein.

In das Feld **SiteActive-Basisverzeichnis** tragen Sie das Verzeichnis ein, in dem der Systemdienst zu den festgesetzten Zeiten auszuführen ist.

Im Abschnitt **SiteActive-Erstellung** tragen Sie die Parameter ein, die für die Ausführung des SiteActive notwendig sind. Trennen Sie dabei die einzelnen Parameter durch Doppelpunkte voneinander. Pflichtparameter sind:

- Angabe des Templates: `TEMPLATE=/ver/zeich/nis/Datei.name`
- Zielseite: `SAVEPAGE=/ver/zeich/nis/datei.html`

Optional besteht die Möglichkeit, weitere Variablen in Form eines Name-Wert-Paares zu definieren. Diese Variablen lassen sich bei der Verarbeitung des SiteActive-Codes als FF-Variablen referenzieren.

Beispiel für einen Eintrag im Feld **SiteActive-Erstellung**:

`TEMPLATE=/movies/index.html:SAVEPAGE=/index.html:`**`foo=bar`**`:myvar=myvalue`

Im Feld **Alarm-Verzeichnis** tragen Sie das Verzeichnis ein, das der Hermes-Daemon für diesen Systemdients überwachen soll.

Wenn Sie mehrere Verzeichnisse überwachen wollen, tragen Sie diese als kommaseparierte Liste in das Feld ein:

```
/verz1,/verz2,/verz3,/verz4/news/
```

Sie benötigen also nur einen Systemdienst für die Überwachung mehrerer Verzeichnisse.

Wichtig:

Die Pfadangaben für `TEMPLATE`, `SAVEPAGE` *und Alarmverzeichnis interpretiert das System immer ausgehend von der Document-Root Ihres* **imperia** *-Systems.*

Wenn Sie bestimmte Verzeichnisse oder Dateien explizit von der Überwachung ausschließen möchten, tragen Sie diese mit vorangestelltem Ausrufezeichen in das Feld ein:

```
/verz1,!/verz1/print.html,/verz2/news,!/verz2/news/archive
```

Der Auslöser für den Systemdienst sind in diesem Fall die `Notification Files` , die im Verzeichnis `/site/incoming` eingehen. Der Daemon Hermes überwacht dieses Verzeichnis und prüft anhand der Pfadangaben, die in den Notifications stehen, ob ein dazu passender Systemdienst existiert. Stimmt die Pfadangabe mit dem Alarm-Verzeichnis eines Systemdienstes überein, stößt er dessen Ausführung an und löscht anschließend das `Notification File`. Lesen Sie hierzu auch den Abschnitt ???.

Klicken Sie auf **Okay**, um den neuen Systemdienst zu speichern, oder auf **Schließen**, um den Dialog zu verlassen, ohne die Einstellungen zu speichern.

Führen Sie die Funktion **Menü**->**System**->**Systemdateien abgleichen** aus, um die geänderten Einstellungen auf die betreffenden Zielsysteme zu übertragen.

2.5.2 Systemdienst ändern

Um einen vorhandenen Systemdienst zu ändern, gehen Sie wie folgt vor:

1. Rufen Sie den Dialog für die Systemdienstkonfiguration auf: **Menü**->**System**->**Systemdienste**.

   Alle bereits eingerichteten Systemdienste sind dort durch einen horizontalen Balken voneinander getrennt in mehreren Abschnitten aufgelistet.

   Für jeden Abschnitt sind die angezeigten Systemdienste individuell sortierbar. Dafür stehen mehrere Sortieroptionen zur Verfügung, die Sie jeweils mit einer Pull-Down-Liste auswählen.

2. Klicken Sie im Abschnitt des zu ändernden Dienstes auf den Link **[edit]**. Es öffnet sich der Dialog **Systemdienst bearbeiten**. Die Einstellungsmöglichkeiten sind die selben wie beim Anlegen eines neuen Systemdiensts
3. Nehmen Sie die gewünschten Änderungen vor.
4. Klicken Sie auf **Okay**, um Ihre Änderungen zu speichern, oder auf **Schließen**, um sie zu verwerfen.
5. Sie gelangen zurück in den Dialog **Systemdienste**, wo Sie Ihre gespeicherten Änderungen kontrollieren können.

Die Änderung eines Systemdienstes ist damit abgeschlossen.

Wichtig:

Neue Systemdienste und Änderungen an einem bestehenden Systemdienst werden erst nach einem Abgleich der Systemdateien aktiv. Führen Sie dazu die Funktion **Menü**->**System**->**Systemdateien abgleichen** *aus.*

2.5.3 Systemdienst löschen

Achtung:

Beim Löschen von Systemdiensten erscheint keine Sicherheitsabfrage.

Um einen vorhandenen Systemdienst zu löschen, gehen Sie wie folgt vor:

1. Rufen Sie den Dialog für die Systemdienstkonfiguration auf: **Menü**->**System**->**Systemdienste**.

   In den einzelnen Abschnitten sind alle vorhandenen Systemdienste aufgelistet.

2. Klicken Sie auf den Link **[delete]** im Abschnitt des zu löschenden Dienstes.

Wichtig:

Um den Systemdienst auch auf den Zielsystemen zu löschen, führen Sie die Funktion **Menü**->**System**->**Systemdateien abgleichen** *aus.*

2.6 Dokumentenhaltung

imperia unterstützt neben der Dokumentenhaltung im Dateisystem auch eine Reihe von SQL-Datenbanken (RDBMS):

kommerzielle Produkte:	Oracle 8i (und höher)
	MS-SQL ab Version 2005
frei verfügbare Produkte:	PostgreSQL 8.X (und höher)
	MySQL ab Version 5.X

Tabelle 2.1. SQL-Datenbanken für die Dokumentenhaltung

Weitere Datenbanken wie Informix oder SyBase können auf Anfrage und nach entsprechenden kostenpflichtigen Anpassungen zur Dokumentenhaltung genutzt werden.

Eine Beschreibung der Tabellen und deren Attribute finden Sie im Anhang in ???.

Für die nachfolgenden Kapitel setzen wir voraus, dass sowohl die Installation des RDBMS-Servers als auch des Clients bereits erfolgreich abgeschlossen wurde.

2.6.1 Unterschiede zum Dateisystem und Einschränkungen

Dokumente auf Zielsystem bearbeiten

Bei einer Dokumentenhaltung in einem RDBMS ist im Gegensatz zur Dokumentenhaltung im Dateisystem eine Einschränkung der Suche über URI-Maske mit regulären Ausdrücken nur mit (Teil-)Strings der URI möglich. Man kann also eine Zeichenfolge eingeben, die der URI enthält. Hierbei sind auch Slashes erlaubt.

Schreibtisch

Bei Dokumentenhaltung in einem RDBMS ist die Sortierung der angezeigten Dokumente nicht nach allen Kriterien möglich. Hierzu gehört zum Beispiel UID. Für einige Produkte ist die Realisierung der Sortierung nach allen Kriterien geplant (z.B.: PostgreSQL), für andere nicht (z.B.: MySQL), da dies große Performance-Einbußen mit sich bringen würde.

2.6.2 Konfiguration in imperia

Nach der Installation ist die Dokumentenhaltung in **imperia** auf das File65-Plug-In eingestellt. Im Menüpunkt **Dokumentenhaltung** der Systemadministration kann ein anderes Plug-In ausgewählt werden. Die verschiedenen Arten der Dokumentenhaltung werden durch unterschiedliche Plug-Ins realisiert.

Folgende Plug-Ins stehen zur Verfügung:

- File65
- MySQL
- PostgreSQL
- ODBC_MSSQL
- Oracle

Der Konfigurationsdialog der Dokumentenhaltung in **imperia** wird über den Menüpunkt **Dokumentenhaltung** im Ordner **System** aufgerufen.

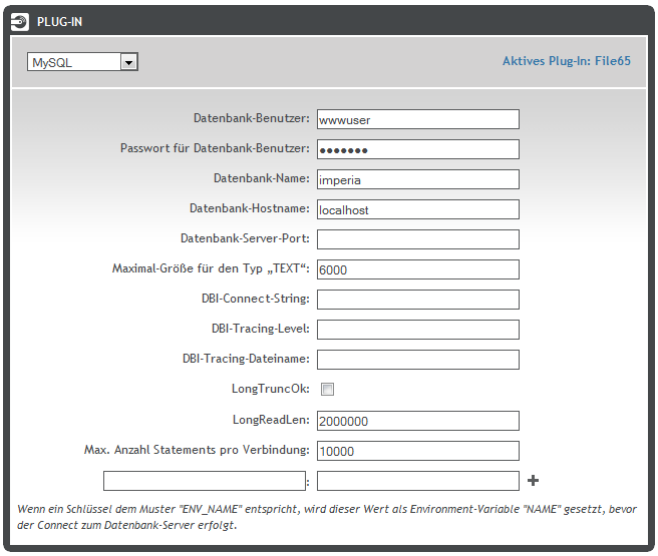


Abb. 2.10: Dialog Dokumentenhaltung

Hinweis:

Die in diesen Dialog eingegebenen Parameter werden in der Datei `/site/config/repositories/default.conf` *gespeichert.*

Der Dialog ist in zwei Abschnitte gegliedert:

Auswahl

In diesem Abschnitt wird das zu verwendende Plug-In über die Drop-Down-Box ausgewählt. Neben der Drop-Down-Box wird das momentan aktive Plug-In angezeigt.

Konfigurationsabschnitt

Dieser Abschnitt muss für jedes Plug-In gesondert bearbeitet werden. Er enthält die Eingabefelder für die Parameter, die für ein Plug-In definiert werden. In der Abbildung werden die Eingabefelder für das PostgreSQL-Plug-In angezeigt.

Ferner können im unteren Teil des Abschnittes benutzerdefinierte Schlüssel hinzugefügt werden. In die Eingabefelder müssen folgende Parameter eingetragen werden:

Datenbank-Name (Pflichteintrag!)

Name der zu verwendenden Datenbank. Diese Datenbank muss manuell angelegt werden.

Datenbank-Benutzer (Pflichteintrag!)

Name des Benutzers, der die Datenbank benutzen darf. Dieser Benutzer muss manuell angelegt werden und die passenden Rechte für die unter „Datenbank-Name“ angegebene Datenbank besitzen.

Passwort für den Datenbank-Benutzer (Pflichteintrag!)

Passwort des Datenbank-Benutzers.

Datenbank-Hostname (Pflichteintrag, wenn kein DBI-Connect-String definiert ist!)

Name des Hosts, auf dem die Datenbank installiert ist. Gegebenenfalls muss die dazugehörige Domäne angegeben werden.

Datenbank-Server-Port

Optional. Diesen Port nur angeben, wenn er sich vom jeweiligen Standard-Port des RDBMS unterscheidet.

Maximal-Größe für den Typ "TEXT"

Optional. Maximal mögliche Eingabe in eine Spalte vom ANSI-SQL-Typ TEXT bzw. für MSSQL-Server VARCHAR. Dieser Wert kann bei von **imperia** unterstützten RDBMS leer bleiben. Je näher dieser Wert aber dem tatsächlich maximal möglichen kommt, desto besser ist die Performance, da dann bestimmte Daten weniger fragmentiert werden.

Maximal-Größe für den Typ "TEXT"

Optional. Maximal mögliche Eingabe in eine Spalte vom ANSI-SQL-Typ TEXT bzw. für MSSQL-Server VARCHAR. Dieser Wert kann bei von **imperia** unterstützten RDBMS leer bleiben. Je näher dieser Wert aber dem tatsächlich maximal möglichen kommt, desto besser ist die Performance, da dann bestimmte Daten weniger fragmentiert werden.

Blocksize

Über die Blocksize kann die Größe der Binärblöcke angegeben werden. Bei Oracle ist der Maximalwert 4000. Für PostgeSQL un MySQL können auch größere Werte verwendet werden.

DBI-Tracing-Level

Dies ist der Grad des Informationsgehalts des DBI-Traces. Je höher die hier eingetragene Ziffer ist, desto mehr Informationen werden in der Trace-Log-Datei gespeichert. Die Standardeinstellung sollte `0` (kein Trace) sein. Für die Entwicklung neuer Datenbank-Plug-Ins empfehlen wir die Einstellung `3`, da die Einstellungen `1` und `2` aufgrund eines internen Problems des DBI-Moduls nicht korrekt funktionieren.

DBI-Tracing-Dateiname

Dies sind der absolute Pfad und Dateiname der Trace-Log-Datei.

LongTruncOk

Sollte nicht gesetzt werden, da im Fehlerfall lange Strings beschnitten werden, ohne dass eine Fehlermeldung auftritt.

LongReadLen

Dies ist die maximale Länge eines Strings, der gelesen werden kann. Defaultwert ist 2 Mio.

Max. Anzahl Statements pro Verbindung

Dieser Punkt gibt die maximale Anzahl der SQL-Statements an, bevor die Verbindung zur Datenbank wieder geschlossen und neu aufgebaut wird.

Zusätzliche Treiber-Optionen (nur PostgreSQL)

Hier können weitere Treiber-Einstellungen für das PostgreSQL-Plug-In getätigt werden.

Debugging-TTY

Dies ist eine zusätzliche Einstellung, auf welchem Screen die Debugging-Informationen ausgegeben werden sollen. Gilt nur für PostgreSQL unter UNIX®.

SID (Pflichteintrag für Oracle)

Dies ist der Name der Oracle-Instanz.

Sonstige Eingabefelder

Mit Hilfe der beiden Eingabefelder können eigene Parameter definiert und gesetzt werden.

Die eingegebene Konfiguration wird gespeichert, sobald der Button **Okay** am unteren Rand der **imperia** -Oberfläche betätigt wird. Der Dialog wird dann im oberen rechten Bereich um die folgenden Links ergänzt:

Tabellen erzeugen

Mit diesem Link werden alle erforderlichen Tabellen automatisch erzeugt. Er öffnet einen weiteren Dialog, in dem der Benutzername und das Passwort des RDBMS-Administrators bzw. eines Benutzer mit den entsprechenden Rechten eingegeben werden muss. Lesen Sie hierzu Abschnitt 2.6.4 **Tabellen erzeugen** auf Seite 21.

SQL-Erzeugungs-Statements anzeigen (Trockenlauf)

Dieser Link dient zum Testen der Konfiguration. **imperia** erzeugt die Tabellen nicht, sondern simuliert eine Erzeugung mit der eingestellten Konfiguration und testet damit die Verbindung zum RDBMS.

Tabellen löschen

Mit diesem Link werden die Tabellen gelöscht. Es wird ein Dialog geöffnet, in den der Benutzername und das Passwort des RDBMS-Administrators bzw. eines Benutzers mit den entsprechenden Rechten eingegeben werden muss. Lesen Sie hierzu Abschnitt 2.6.5 **Tabellen löschen** auf Seite 21.

SQL-Lösch-Statements anzeigen (Trockenlauf)

Dieser Link dient zum Testen der Konfiguration. **imperia** löscht keine Tabellen, sondern simuliert das Löschen mit der eingestellten Konfiguration und testet damit die Verbindung zum RDBMS.

2.6.2.1 PostgreSQL

Das PostgreSQL-Plug-In bietet eine Schnittstelle zu einer PostgreSQL-Datenbank. Voraussetzung für **imperia** ist die PostgreSQL-Version 8.x.

2.6.2.2 ODBC_MSSQL

Das ODBC_MSSQL-Plug-In bietet eine Schnittstelle zu einer MSSQL-Datenbank über ODBC.

2.6.2.3 Oracle

Das Oracle-Plug-In bietet eine Schnittstelle zu einer Oracle-Datenbank.

2.6.3 Konfiguration der Dokumentenhaltungsschnittstelle

Voraussetzung für die Konfiguration der Dokumentenhaltung in einer SQL-Datenbank ist die bereits erfolgte Installation folgender Perl-Module, die nicht Bestandteil der Standard-Perl-Distribution sind:

2.6.3.1 Notwendige Perl-Module

DBI

Dieses Modul (Dateiname DBI.pm) kann für UNIX®-Betriebssysteme vom CPAN heruntergeladen und installiert werden. Für Windows®-Betriebsysteme wird es von ActiveState bereitgestellt und kann mit Hilfe von `ppm` direkt heruntergeladen und installiert werden.

Achtung:

Verwenden Sie unter keinen Umständen die DBI-Version 1.26, da sie Fehler enthält, die einen reibungslosen Betrieb verhindern.

Hinweis:

Wenn Sie eine Oracle-Datenbank verwenden wollen, muss mindestens die DBI-Version 1.23 installiert werden. Eventuell muss die Datei `DBI.pm` *angepasst werden, um volle Kompatibilität mit* **imperia** *zu erzielen:*

Fügen Sie im BEGIN-Block (z.B.: bei Version 1.28 von DBI.pm in Zeile 11) zur Versionsnummer Anführungszeichen hinzu:

```
$DBI::Version = "1.28"; #...
```

RDBMS-spezifische Treiber

Abhängig von der Datenbank, die verwendet werden soll, muss ein weiterer spezifischer Treiber installiert werden. Die Treiber erhält man auf die gleiche Weise wie das DBI-Modul (CPAN bzw. ppm).

Die Treiber haben folgende Bezeichnungen:

RDBMS	Modulname
PostgreSQL	DBD::Pg
Oracle	DBD::Oracle
MySQL	DBD::MySQL
MSSQL	DBD::ODBC

Tabelle 2.2. RDBMS spezifische Treiber

Tipp

Ein umfassendes Repository mit Perl-Modulen für ActiveState Perl finden Sie unter

```
http://theoryx5.uwinnnipeg.ca/ppmpackages (Perl-Version 5.6.X)
http://theoryx5.uwinnnipeg.ca/ppms (Perl-Version 5.8.X)
```

Wenn Sie die Module nicht direkt mit ppm auf dem betreffenden System installieren können oder möchten, besteht auch die Möglichkeit, diese zunächst auf einem anderen Rechner herunterzuladen.

*In den o.a. Verzeichnissen finden Sie jeweils die *.ppd-Datei für die Installation der einzelnen Module. Die eigentlichen Module liegen als Tarball unter folgenden URLs:*

```
http://theoryx5.uwinnnipeg.ca/ppmpackages/x86 (Perl-Version 5.6.X)
http://theoryx5.uwinnnipeg.ca/ppms/x86 (Perl-Version 5.8.X)
```

Im Folgenden finden Sie eine Aufstellung der Treiber, die auf dem CPAN zur Verfügung stehen (xxxx steht für die Versionsnummer):

RDBMS	DBD-Treiber
PostgreSQL	DBD-Pg-xxxx.tar.gz
Oracle	DBD-Oracle-xxxx.tar.gz
MySQL	DBD-mysql-xxxx.tar.gz
MSSQL	DBD-ODBC-xxxx.tar.gz

Tabelle 2.3. CPAN-Module

2.6.3.2 PostgreSQL und Linux

1. Laden Sie das Modul DBD-Pg-xxxx.tar.gz vom CPAN herunter.
2. Melden Sie sich mit root-Rechten am Rechner an und verschieben Sie das heruntergeladene Modul in ein temporäres Verzeichnis.
3. Entpacken Sie das Modul mit folgendem Befehl:

   ```
   tar -zxvf DBD-Pg-xxxx.tar.gz
   ```

   Es wird das Verzeichnis DBD-Pg-xxxx erzeugt.
4. Wechseln Sie in das gerade erzeugte Verzeichnis und generieren Sie das Makefile mit folgendem Befehl:

```
perl Makefile.pl
```

5. Folgen Sie der Standard-Installationsprozedur:

```
make
make test
make install
```

6. Die Installation ist abgeschlossen, wenn die folgenden beiden Dateien existieren:

```
/usr/lib/perl5/site_perl/<ihre Perlversion>/i586-linux/DBD/Pg.pm
/usr/lib/perl5/site_perl/<ihre Perlversion>/i586-linux/Pg.pm
```

Installation von PostgreSQL unter SuSE 7.1 (oder höher)

Wird eine SuSE-Distribution der Version 7.1 oder höher verwendet, kann die mit SuSE gelieferte PostgreSQL-Version verwendet werden.

Laden und entpacken Sie anschließend den Datenbank-Treiber wie oben beschrieben. Bevor der Treiber unter SuSE installiert werden kann, müssen zunächst das SuSE-Paket `postgres-devel` installiert und folgende Umgebungsvariablen `POSTGRS_LIB` und `POSTGRES_INCLUDE` gesetzt werden.

Beispiele:

```
thomas@work:/tmp/DBPD-Pg-1.01 > export POSTGRES_INCLUDE=/usr/include/pgsql/
thomas@work:/tmp/DBPD-Pg-1.01 > export POSTGRES_LIB=/usr/lib/
```

Danach kann die Installation wie oben beschrieben durchgeführt werden. Legen Sie anschließend einen Benutzer für die Datenbank an. Loggen Sie sich als Superuser (Benutzer `postgres`) der Datenbank ein und legen Sie einen neuen Benutzer mit dem Recht, eine Datenbank zu erzeugen, an.

Achtung:

Der Superuser der Datenbank erhält bei der Installation ein dem Autor unbekanntes, nicht standardisiertes Passwort. Ändern Sie dieses Passwort, um neue Datenbank-Benutzer anlegen zu können.

2.6.3.3 PostgreSQL und Windows

- Starten Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install DBD-Pg
```

  Nach erfolgreicher Installation können Sie die Konfiguration und die Erzeugung der Tabellen in **imperia** vornehmen. Eventuell muss die Datei `DBI.pm` in den Perlpfad aufgenommen werden.

Wenn Sie für ppm kein Repository eingerichtet haben, in dem dieses Modul verfügbar ist, können Sie es aus dem in Abschnitt 2.6.3.1, „RDBMS-spezifische Treiber“ erwähnten Repository herunterladen.

2.6.3.4 Oracle und Linux

Bevor der Oracle-Treiber installiert werden kann, muss der Oracle-Client installiert werden. Hierzu verwendet man die gelieferte Oracle-CD, startet das Installationsprogramm, wählt den Client aus und folgt den Anweisungen. Lediglich der Port, den Sie bei Ihrem Systemadministrator erfragen können, muss eventuell angepasst werden.

Nach erfolgreicher Installation des Oracle-Clients kann der Treiber installiert werden.

1. Laden Sie das Modul DBD-Oracle-xxxx.tar.gz vom CPAN herunter.
2. Melden Sie sich mit root-Rechten am Rechner an und verschieben Sie das heruntergeladene Modul in ein temporäres Verzeichnis.
3. Entpacken Sie das Modul mit folgendem Befehl:

   ```
   tar -zxvf DBD-Oracle-xxxx.tar.gz
   ```

   Es wird das Verzeichnis `DBD-Oracle-xxxx` erzeugt.
4. Wechseln Sie in das gerade erzeugte Verzeichnis und generieren Sie dass Makefile mit folgendem Befehl:

   ```
   perl Makefile.pl
   ```

5. Folgen Sie der Standard-Installationsprozedur:

   ```
   make
   make test
   make install
   ```

2.6.3.5 MySQL und Windows®

- Starten Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

  ```
  ppm install DBD-mysql
  ```

  Nach erfolgreicher Installation können Sie die Konfiguration und die Erzeugung der Tabellen in **imperia** vornehmen. Eventuell muss die Datei `DBI.pm` in den Perlpfad aufgenommen werden.

2.6.3.6 MySQL und Linux

Bei der Installation von MySQL müssen Sie auf die Version MySQL 5.X zurückgreifen, die Mindestvoraussetzung für **imperia** ist.

Nach erfolgreicher Installation von MySQL installieren Sie den DBD-mysql-xxxx-Treiber:

1. Melden Sie sich mit root-Rechten am Rechner an und verschieben Sie das heruntergeladene Modul in ein temporäres Verzeichnis.
2. Das Modul wird mit folgendem Befehl entpackt:

   ```
   tar -zxvf DBD-mysql-xxxx.tar.gz
   ```

   Es wird das Verzeichnis DBD-mysql-xxxx erzeugt.
3. Wechseln Sie in das erzeugte Verzeichnis und generieren Sie das Makefile mit folgendem Befehl:

   ```
   perl Makefile.pl
   ```

4. Folgen Sie der Standard-Installationsprozedur:

   ```
   make
   make test
   make install
   ```

2.6.3.7 SQL-Server und Windows®

1. Installieren Sie den ODBC-Treiber von ActiveState mit Hilfe von `ppm` .

   ```
   ppm install DBD-ODBC
   ```

2. Richten Sie anschließend unter

   ```
   Start/Einstellungen/Systemsteuerung/Verwaltung/Datenquellen ODBC
   ```

   den SQL-Server als System-DSN ein.

3. Beantworten Sie die Frage "Wie soll SQL-Server die Authentizität des Benutzernamens bestätigen?" mit der zweiten Antwort "Mit SQL-Server Authentifizierung anhand des vom Benutzer eingegebenen Benutzernamens und Kennworts".

   Der Datenquellen-Name, der hier vergeben wird, muss in der Konfiguration der **imperia** -Dokumentenhaltung als **Database Name** angegeben werden.

2.6.4 Tabellen erzeugen

Alle notwendigen Tabellen werden automatisch erzeugt. Beim Klick auf den Link **Tabellen erzeugen** erscheint ein weiterer Dialog, in den der Name und das Passwort des RDBMS-Administrators bzw. eines Benutzers mit entsprechenden Rechten eingegeben werden muss. Im Feld **Aktion** muss der Eintrag **Tabellen erzeugen** ausgewählt sein.

Wurden alle Angaben gemacht, können die Tabellen mit einem Klick auf den Button **Okay** erzeugt werden. Nach erfolgreicher Durchführung wird eine Liste aller erzeugten Tabellen angezeigt.

2.6.5 Tabellen löschen

Alle **imperia** -Tabellen werden automatisch gelöscht. Dabei gehen alle Daten verloren. Ein Klick auf den Link **Tabellen löschen** öffnet einen weiteren Dialog, in den der Name und das Passwort des RDBMS-Administrators bzw. eines Benutzers mit entsprechenden Rechten eingegeben werden muss.

Hinweis:

Bleiben die Felder **Benutzername** *und* **Passwort** *leer, versucht* **imperia** *die Tabellen mit dem bereits in der Dokumentenhaltungskonfiguration angegebenen Benutzernamen/Passwort zu erzeugen. Hat dieser Benutzer die entsprechenden Rechte nicht, können die Tabellen nicht angelegt werden.*

Im Feld **Aktion** muss der Eintrag **Tabellen löschen** ausgewählt sein.

Wurden alle Angaben gemacht, können die Tabellen mit einem Klick auf den Button **Okay** gelöscht werden. Nach erfolgreicher Durchführung wird eine Liste aller gelöschten Tabellen angezeigt.

2.7 One-Click-Edit

Mit One-Click-Edit (OCE) können Sie ein veröffentlichtes Dokument, das Sie auf einem Zielsystem betrachten direkt von dort aus zum Bearbeiten öffnen. Im Hintergrund sorgt **imperia** für die Einhaltung des zugehörigen Workflows und der Freigabemechanismen. Sie brauchen lediglich eine Browser-Erweiterung zu installieren, um OCE nutzen zu können.

imperia erkennt, ob ein Benutzer bereits angemeldet ist und bietet bei entsprechender Konfiguration automatisch die Login-Maske an. Im Folgenden wird die Konfiguration der verfügbaren Funktionen der OCE-Toolbar beschrieben.

Innerhalb eines **imperia** -Dokuments werden bei entsprechender Programmierung des Templates die NodeID und die System-ID verschlüsselt gespeichert. Hierzu wird entweder ein eigener Namespace oder ein META-Tag verwendet. Bei der META-Tag-Variante ist außerdem die Speicherung von Pfad und Dateiname der betreffenden Seite möglich. So lassen sich beispielsweise Copy-Seiten unabhängig vom Master-Dokument oder anderen Kopien vom Livesystem entfernen. Lesen Sie hierzu auch den Abschnitt "**One-Click-Edit**" im Programmierhandbuch und Abschnitt 17.18.1 **Variable ONECLICKEDIT** auf Seite 231.

Bei installierter OCE-Toolbar, gleicht One-Click-Edit nach der Anmeldung des Benutzers am zugehörigen Entwicklungssystem bei jedem Seitenaufruf automatisch die Benutzer-Rechte sowie Dokumenten-Informationen mit dem Develop-Server ab, wenn die aufgerufene Seite von diesem System stammt. Der Benutzer kann immer nur die Funktionen der Toolbar aufrufen bzw. sehen, zu deren Ausführung er für das jeweilige Dokument berechtigt ist.

Wichtig:

Beachten Sie, dass beim Kopieren eines Systems, etwa zu Testzwecken, die SysID ebenfalls mitkopiert wird. Dies führt zu Problemen, wenn Benutzer auf dem Testsystem Content freischalten oder mit One-Click-Edit Dokumente bearbeiten. Da die kopierte SysID eingelesen wird, arbeiten die Benutzer nicht auf dem Testsystem sondern auf dem Produktivsystem, von dem die Kopie angefertigt wurde.

Damit dieser Fehler nicht auftritt, müssen Sie in der Kopie eine neue SysID erzeugen. Gehen Sie hierzu folgendermaßen vor:

1. *Löschen Sie die Datei* `site/secure/sysid` *in Ihrem Testsystem. Melden Sie sich anschließend an dem System an. Dabei wird automatisch eine neue SysID generiert.*
2. *Reparsen Sie den kompletten Content Ihres Testsystems, damit die neue SysID in die Dokumente eingebunden wird.*
3. *Nun können Sie gefahrlos auf dem Testsystem arbeiten.*

OCE wird im Dialog **One-Click-Edit-Konfiguration** konfiguriert, der über den Menüpunkt **One-Click-Edit** im Ordner **System** geöffnet wird. Der Dialog enthält für jede mögliche OCE-Funktion einen eigenen Abschnitt.

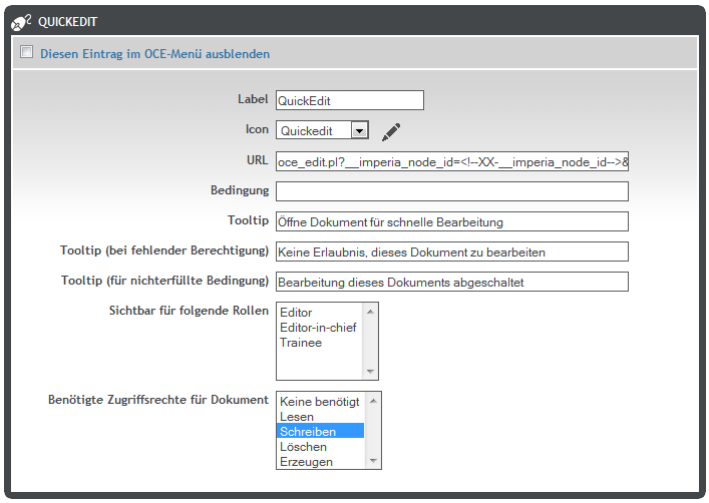


Abb. 2.11: Dialog OCE-Konfiguration

Der Dialog gliedert sich in zwei Abschnitte. Im oberen Teil wird unter anderem eingestellt, welche Rollen zur Verwendung dieser OCE-Funktion berechtigen. Ferner wird eingestellt, welche Dokumenten-Zugriffsrechte benötigt werden.

Im unteren Abschnitt werden die URL, eine Bedingung sowie verschiedene Tooltips eingegeben.

Die Elemente des Dialogs bedeuten im Einzelnen:

Hide
: Mit dieser Checkbox wird die Funktion global aus- bzw. eingeschaltet. Wird die Checkbox markiert, steht die Funktion nicht zur Verfügung.

Icon
In dieser Liste wird das Icon ausgewählt, das in der OCE-Toolbar angezeigt werden soll.

Label
In dieses Eingabefeld wird der Titel der Funktion eingegeben. Der Titel erscheint in der OCE-Toolbar neben dem Icon.

Sichtbar für folgende Rollen
In diesem Feld werden die Rollen markiert, die zur Verwendung der OCE-Funktion berechtigen. Mehrere Rollen werden mit Hilfe der Strg-Taste markiert.

Benötigte Zugriffrechte für Dokument
In diesem Feld werden die Zugriffsrechte für ein Dokument bestimmt, die der Benutzer mindestens haben muss, um die Funktion ausführen zu können. In der Hilfe werden die Rechte für eine Funktion empfohlen.

URL
In diesem Feld wird die URL eingegeben, den die OCE-Toolbar an das Produktionssystem schicken soll. Er besteht aus dem Namen des Skripts sowie verschiedenen Parametern. Lesen Sie hierzu auch Abschnitt 2.7.2 **OCE-Skript hinzufügen** auf Seite 23.

Hilfe
Dieser Button zeigt einen Hilfetext zu der ausgewählten OCE-Funktion. Die Hilfe basiert auf den PODs der entsprechenden **imperia** -Skripte.

Bedingung
Definieren Sie eine optionale Bedingung, die erfüllt sein muss, damit die OCE-Funktion in der OCE-Toolbar erscheint. Die Notation erfolgt in der Template-Syntax der IF-Abfragen, z.B.: "<!--XX-directory-->" NE "/no_quick_edit").

Tooltip
Hier wird der Text eingegeben, der angezeigt wird, wenn der Benutzer den Mauszeiger über dem Icon in der OCE-Toolbar hält.

Tooltip (bei fehlender Berechtigung)
Hier wird der Text eingegeben, der angezeigt wird, wenn der Benutzer den Mauszeiger über dem Icon in der OCE-Toolbar hält, er aber nicht über die notwendigen Rechte verfügt.

Tooltip (für nichterfüllte Bedingung)
Hier wird der Text eingegeben, der angezeigt wird, wenn der Benutzer den Mauszeiger über dem Icon in der OCE-Toolbar hält, jedoch die eingestellte Bedingung nicht erfüllt ist.

2.7.1 One-Click-Edit und Copy-Seiten

Normalerweise arbeitet One-Click-Edit auf Dokumentenbasis. Das bedeutet für Dokumente mit Copy-Pages, dass eine OCE-Funktion, die Sie auf einer Copy-Seite ausführen, sich immer auf das gesamte Dokument auswirkt. Beispielsweise löschen Sie also im Normalfall mit der Funktion **Live-Löschen** nicht nur die jeweilige Kopie vom Livesystem, sondern auch alle anderen existierenden Kopien des Dokuments. Durch eine besondere Einbindung der OCE-Informationen ins Template eines Dokuments oder durch eine Variable zur Systemkonfiguration können Sie dieses Verhalten auf eine dateibasierte Verarbeitung umstellen. Für weitere Einzelheiten hierzu lesen Sie bitte Abschnitt 17.18.1 **Variable ONECLICKEDIT** auf Seite 231 und den Abschnitt **One-Click-Edit** in der Syntaxreferenz im Kapitel **Templates** des Programmierhandbuchs.

2.7.2 OCE-Skript hinzufügen

Neben den mitgelieferten OCE-Skripten können auch andere **imperia** -Skripte oder eigene Skripte als OCE-Skript verwendet werden. OCE-Skripte werden im CGI-BIN-Verzeichnis erwartet. Als Namenskonvention gilt, dass die **imperia** -OCE-Skripte mit `oce_` beginnen müssen.

Am unteren Ende des Dialogs **One-Click-Edit-Konfiguration** befindet sich ein leerer Abschnitt, in den ein weiteres OCE-Skript eingetragen werden kann. Sobald eine neue Skript-Konfiguration gespeichert wurde, erscheint automatisch ein neuer leerer Abschnitt. Am Beginn dieses Kapitels werden die einzelnen Felder des Dialogs und die notwendigen Eingaben beschrieben.

Wir möchten hier auf das Eingabefeld **URL** eingehen, in dem die URL festgelegt wird, den der entsprechende Button der OCE-Toolbar übergibt. Es können auch Links eingegeben werden, die nicht ein **imperia** -Produktionssystem ansprechen. Zum Beispiel zu einem Übersetzungsdienst oder einer Suchmaschine.

Eine URL, die das OCE-Skript übergibt, besteht aus dem Namen des Skripts und verschiedenen Parametern. Ein wichtiger Parameter ist die **imperia** -NodeID, über die das Dokument vom Produktionssystem identifiziert wird.

2.7.3 imperia OCE-Skripte

Folgende OCE-Funktionen sind verfügbar:

2.7.3.1 oce_edit

Dieses Skript öffnet das Dokument zum Editieren. Befindet sich das Dokument nicht im Workflow, wird die letzte gespeicherte Version aus dem Archiv importiert, bevor es zum Editieren geöffnet wird. Als Dokumenten-Zugriffsrechte sollte **Lesen** und **Schreiben** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Die NodeID des Dokuments.

finished

Wird dieser Parameter als `true` (oder `1`) übergeben, wird die letzte **beendete** Version und nicht die zuletzt gespeicherte Version aus dem Archiv importiert.

__imperia_quick_edit

Wird dieser Parameter gesetzt, werden alle Workflow-Plug-Ins, die QuickEdit nicht unterstützen, ausgelassen.

2.7.3.2 oce_clone

Dieses Skript erzeugt einen fast identischen Klon eines **imperia** -Dokuments. Das neue Dokument unterscheidet sich zumindest in den Meta-Variablen `__imperia_node_id`, `__imperia_created` und `__imperia_modified`. Auch der Wert von anderen Meta-Variablen kann mit Hilfe passender Parameter verändert werden. Als Dokumenten-Zugriffsrechte sollte **Lesen**, **Schreiben** und **Erzeugen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Die NodeID des Dokuments.

__imperia_created, __imperia_modified

Diese beiden Variablen enthalten die aktuelle Zeit in Sekunden seit der Epoche (Sekunden seit 00:00:00, 01.01 1904 GMT für MacOS und 00:00:00, 01.01.1970 für UNIX®- und Windows®-Betriebssysteme).

__imperia_uid

Enthält die Benutzer-ID des aktuell angemeldeten Benutzers.

__imperia_meta_expand

Sofern die Meta-Variable `directory` nicht vorhanden ist, enthält dieser Parameter den Wert `1`, unabhängig von seinem vorhergehenden Wert. Dieser Parameter wird vom MetaEdit-Plug-In verarbeitet. Ist er vorhanden und nicht `0`, wird `<!--XX-METAMODE-->` zu `true` expandiert. Der Parameter wird nach der ersten Verarbeitung gelöscht. Infolgedessen wird ein neues Dokument, wenn der Parameter einen wahren Wert hat, vom MetaEdit-Plug-In wie ein gerade erzeugtes Dokument behandelt. Dies wird meist für die Expansion von `<!-- XX-count-->` in Metadateien verwendet.

Hinweis:

Unter normalen Bedingungen sollte das Metafeld `directory` *immer gelöscht werden, bevor ein Dokument geklont wird. Denn dann wird es durch die neu zugeordnete Rubrik initialisiert.*

__imperia_parent

Dieser Parameter enthält die NodeID des Eltern-Dokuments.

__imperia_children

Dieser Parameter enthält die NodeID des Kind-Dokuments, die an die Liste aller Kind-Dokumente eines Eltern-Dokuments(!) angehängt wird.

2.7.3.3 oce_erase

Mit diesem Skript wird ein **imperia** -Dokument komplett gelöscht. Es wird von allen Zielsystemen und aus dem Produktionssystem gelöscht. Ferner werden alle Archivinformationen und alle zugehörigen Objekte (Bilder etc.) gelöscht. Als Dokumenten-Zugriffsrechte sollte **Lesen**, **Schreiben**, **Erzeugen** und **Löschen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

force

Wird dieser Parameter auf einen wahren Wert gesetzt, erfolgt das Löschen ohne Bestätigungsdialog. Anderenfalls erscheint eine Liste aller gelöschten Objekte.

2.7.3.4 oce_delete

Mit oce_delete können Sie ein Dokument von allen Zielsystemen löschen. Das Skript kann folgende Parameter entgegennehmen:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

directory

Dies ist das Verzeichnis auf dem Zielsystem.

filename

Dies ist der Dateiname des Dokuments im Verzeichnis des Zielsystems.

copies

Wenn Sie diesen Parameter auf einen wahren Wert setzen, werden auch alle Kopien des Dokumentes entfernt.

del_develop

Wenn Sie diesen Parameter auf einen wahren Wert setzen, wird die Löschaktion auch auf dem Produktionssystem ausgeführt.

2.7.3.5 oce_history

Dieses Skript zeigt die Dokumenten-History an. Als Dokumenten-Zugriffsrechte sollte **Lesen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

2.7.3.6 oce_info

Dieses Skript zeigt eine Reihe von Informationen zum Dokument an. Als Dokumenten-Zugriffsrechte sollte **Schreiben** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

2.7.3.7 oce_linkcheck

Dieses Skript überprüft alle internen und externen Links in einem Dokument. Als Dokumenten-Zugriffsrechte sollte **Lesen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

finished

Wird dieser Parameter auf `1` (true) gesetzt, wird die letzte beendete Version überprüft und nicht, wie standardmäßig, die letzte bearbeitete Version.

__imperia_quick_edit

Wird dieser Parameter gesetzt, werden alle Workflow-Plug-Ins, die QuickEdit nicht unterstützen, ausgelassen.

2.7.3.8 oce_reparse

Dieses Skript parst das Dokument neu, so dass Änderungen am Template übernommen werden. Optional kann das Freischalten auf alle Zielsysteme erzwungen werden. Als Dokumenten-Zugriffsrechte sollte **Lesen**, **Schreiben** und **Erzeugen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

finished

Wird dieser Parameter auf 1 (true) gesetzt, wird die letzte beendete Version als Basis für das Reparsen herangezogen und nicht, wie standardmäßig, die letzte bearbeitete Version.

publish

Wird dieser Parameter auf einen wahren Wert gesetzt, werden erfolgreich reparste Dokumente automatisch auf die Zielsysteme freigeschaltet. Dies geschieht jedoch nur, wenn das Dokument vorher schon einmal freigeschaltet wurde.

2.7.3.9 oce_unpublish

Dieses Skript entfernt das Dokument von allen Zielsystemen. Als Dokumenten-Zugriffsrechte sollte **Löschen** und **Erzeugen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

directory

Dies ist das Verzeichnis auf dem Zielsystem.

filename

Dies ist der Dateiname des Dokuments im Verzeichnis des Zielsystems.

2.8 Allgemeine Einstellungen

Über den Dialog **Allgemeine Einstellungen** werden verschiedene systemweite Einstellungen vorgenommen.

Abb. 2.12: Dialog Allgemeine Einstellungen

2.8.1 Konfiguration

Dieser Abschnitt dient der Navigation durch den Dialog. Mit Hilfe der Links wird auf die einzelnen Abschnitte zugegriffen.

2.8.2 Cookie-Lebensdauer

Tragen Sie hier die Gültigkeitsdauer des **imperia** -Cookies in Sekunden ein. Der voreingestellte Wert ist 7200 Sekunden. Führt ein Anwender für einen Zeitraum, der länger als diese Einstellung ist, keine Aktion aus, muss er sich erneut am System anmelden.

2.8.3 Dokumenten-History

In diesem Abschnitt legen Sie die Einstellungen für die Anzeige der Dokumenten-History fest. Folgende Einstellungsmöglichkeiten stehen Ihnen zur Verfügung:

Anzeigelänge der Dokumenten-History
: Die Anzahl der angezeigten Einträge. Setzen Sie den Wert auf Null, um das Feature zu deaktivieren. Dadurch wird grundsätzlich die vollständige History angezeigt.

Sichtbarkeit für Datum und Uhrzeit in der Dokumenten-History
: Legen Sie fest, wer Datum und Uhrzeit in der Dokumenten-History sehen kann. Zur Auswahl stehen:

- everybody - Alle Rollen.
- nobody - Die Anzeige von Datum und Uhrzeit ist generell unterdrückt.
- superuser - Nur Benutzer mit Zugriff auf die Superuser-Rolle sehen Datum und Uhrzeit.

2.8.4 Java-VM

In diesem Abschnitt wird die Java VirtualMachine konfiguriert. Folgende Einstellungen können vorgenommen werden:

Java Executable
: Geben Sie in dieses Feld den Pfad zum Java-Interpreter oder zum Java-Wrapper-Skript ein (z.B. `/usr/lib/ java/bin/java`).

Classpath
: Tragen Sie in dieses Feld die Classpath-Einstellungen für die Java VirtualMachine auf dem **imperia** -Server ein. Lassen Sie dieses Feld im Zweifelsfall leer.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

2.8.5 Link-Checker

Dieser Abschnitt konfiguriert den Link-Checker für externe Links. Über die Checkbox **Link-Check mit HTTP-Requests aktivieren** wird die Funktion aktiviert. Falls in Ihrem Netzwerk ein Proxy-Server verwendet wird, können entsprechende Einstellungen in den einzelnen Eingabefeldern getätigt werden.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

2.8.6 Superuser Authentifizierungs-Plug-In

Markieren Sie die Checkbox, um das System in den Wartungsmodus mit Superuser-Zugriff zu versetzen. Lesen Sie hierzu auch Abschnitt 2.2.2 **Wartungsmodus mit Superuser-Zugriff** auf Seite 6

2.8.7 System-Sprache

In diesem Abschnitt wird die System-Sprache und das System-Charset eingestellt.

Die Standardeinstellung hierfür ist Englisch. Mit Hilfe des Pull-Down-Menüs können andere Einstellungen ausgewählt werden.

Unabhängig von der ausgewählten Sprache werden sämtliche Daten im Zeichensatz UTF-8 gespeichert und ausgeliefert. Das gilt sowohl für alle Bildschirmausgaben als auch für alle Daten, die von Benutzern eingegeben wurden.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

2.8.8 Template-Prozessor

Standardmäßig werden Skripte, Layer/ILayer, Frames und Formulare im PREVIEW und im EDIT-Modus nicht unterstützt. In diesem Abschnitt kann dieses Verhalten geändert werden. Markieren Sie die entsprechenden Checkboxen.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

2.9 Konfiguration des Systems über die Benutzeroberfläche

imperia bietet Ihnen eine Schnittstelle zur Änderung sämtlicher Konfigurationsdateien. So brauchen Sie keinen Zugriff auf Dateisystemebene mehr, um Einstellungen des Systems zu ändern, die nicht direkt über die Benutzeroberfläche erreichbar sind.

Diese Funktion ist nur für Benutzer mit Superuser-Berechtigung zugänglich. Die Schnittstelle rufen Sie über den Menüpunkt *Menü -> System -> Konfiguration* auf.

Sie gelangen zu einer Übersicht der editierbaren Konfigurationsdateien:

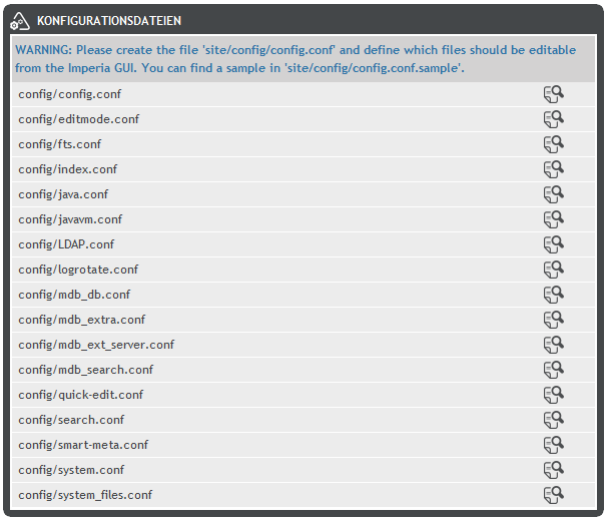
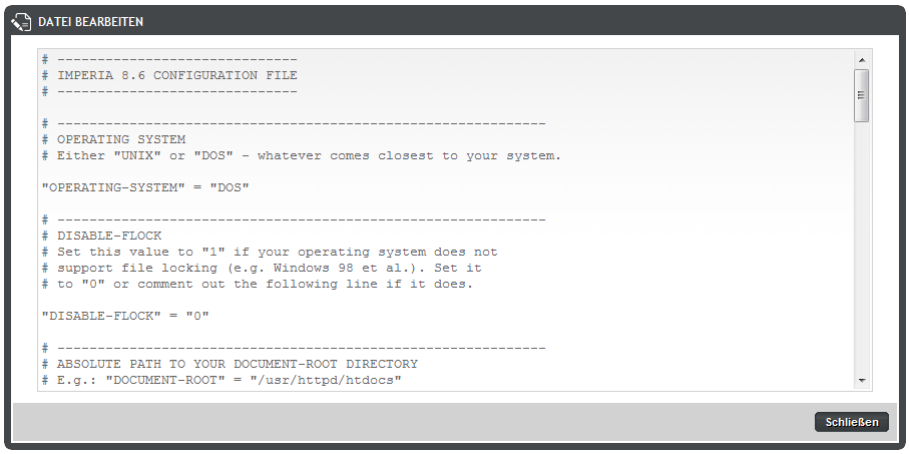


Abb. 2.13: Übersicht der editierbaren Konfigurationsdateien

Klicken Sie zur Auswahl auf eine Zeile. Wenn Sie das Recht haben die Datei zu bearbeiten, dann wird die Datei im Editor geöffnet. Wenn Sie nur das Leserecht auf diese Datei haben, wird der Inhalt der Datei bei der Auswahl angezeigt. ein bearbeiten ist dann nicht möglich. Dateien für die Sie weder Lese- noch Schreibrechte haben, werden nicht zur Auswahl angezeit.

Zur Bearbeitungsansicht einer Konfigurationsdatei gelangen Sie auch, wenn Sie in der Übersicht auf das Icon neben dem Dateinamen klicken. Ein Beispiel für die Bearbeitungsansicht zeigt die nächste Abbildung:


DATEI BEARBEITEN

IMPERIA 8.6 CONFIGURATION FILE

OPERATING SYSTEM
Either "UNIX" or "DOS" - whatever comes closest to your system.
"OPERATING-SYSTEM" = "DOS"

DISABLE-FLOCK
Set this value to "1" if your operating system does not
support file locking (e.g. Windows 98 et al.). Set it
to "0" or comment out the following line if it does.
"DISABLE-FLOCK" = "0"

ABSOLUTE PATH TO YOUR DOCUMENT-ROOT DIRECTORY
E.g.: "DOCUMENT-ROOT" = "/usr/httpd/htdocs"
Schließen

Abb. 2.14: Der Editor für Konfigurationsdateien

Der Editor gleicht in seiner Funktion den Editoren für Flexmodule, Codeincludes und Dynamische Module.

2.9.1 Konfiguration der Schnittstelle

Per Default sind sämtliche Konfigurationen editierbar. Sie können diese Auswahl einschränken, indem Sie die Datei `config.conf` editieren.

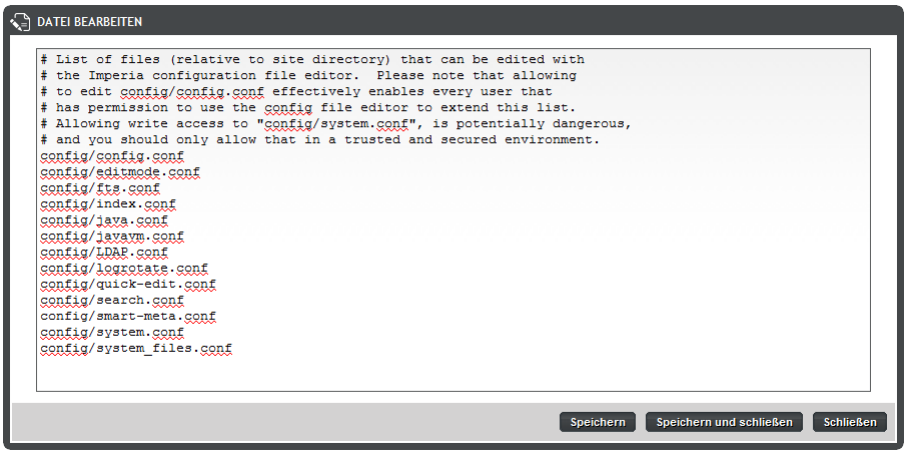


Abb. 2.15: Die Datei `config.conf`

Diese Datei enthält eine Liste der Konfigurationsdateien, die über die Benutzeroberfläche bearbeitet werden können. Um eine Datei aus dieser Liste zu entfernen, löschen Sie den Dateinamen aus der Liste oder setzen Sie das Kommentarzeichen '#' davor.

Umgekehrt können Sie eine Datei editierbar machen, indem Sie sie der Liste hinzufügen oder ein ggf. vorhandenes Kommentarzeichen vor dem Dateinamen entfernen.

2.10 Konfiguration der Link-Konsistenzprüfung

Die Konfiguration der Link-Konsistenzprüfung nehmen Sie in der Datei `linkmanager.conf` im Verzeichnis `site/config` Ihrer **imperia** -Installation vor. Sie können die Links auf mehreren Systemen prüfen lassen. Die Einstellungen für ein System notieren Sie immer innerhalb eines Blocks, der durch die Tags

```
<SERVER MeinServer>

[Einstellungen für das System]

</SERVER>
```

eingeschlossen wird. Anstelle des Platzhalters `MeinServer` tragen Sie eine innerhalb der Konfigurationsdatei eindeutige Bezeichnung für das betreffende System ein.

Diese werden immer nach dem nachfolgenden Schema notiert:

```
Direktive = "Wert"
```

Im Folgenden eine Erläuterung der einzelnen Direktiven für die Konfiguration:

SERVERNAME

Tragen Sie hier einen Namen für das System ein. Dieser Name steht in der Titelzeile des Berichtes mit den Ergebnissen der Link-Konsistenzprüfung für das betreffende System.

```
SERVERNAME = "Livesystem"
```

SERVER_URL

Tragen Sie hier die URL des Systems ein, auf die die Konsistenzprüfung HTTP-Requests zur Prüfung der Links ausführen soll.

```
SERVER_URL = "http://www.myserver.de"
```

SETTYPE_HTML

Geben Sie Dateien an, die die Konsistenzprüfung dem Typ HTML zugehörig einstufen soll. Diese Angabe wird als regulärer Ausdruck auf die Dateinamen angewendet. Alternative Muster werden durch das Pipe-Zeichen ("|") voneinander abgetrennt.

```
SETTYPE_HTML = "\..?htm.?|\..?php"
```

SETTYPE_CSS

Diese Direktive enthält eine Auflistung der Dateien, die als Cascading-Stylesheet-Dateien behandelt werden sollen. Geben Sie die entsprechenden Dateiendungen als regulären Ausdruck an. Die alternativen Dateiendungen werden durch das Pipe-Zeichen ("|") voneinander abgetrennt.

```
SETTYPE_CSS = "\.css/i"
```

EXCLUDE_FILE

Tragen Sie hier einen regulären Ausdruck für die Dateien ein, die von der Link-Konsistenzprüfung nicht kontrolliert werden sollen. Dateien, deren Name auf das Muster passt, werden ausgelassen.

```
EXCLUDE_FILE = "^print"
```

EXCLUDE_DIR

Tragen Sie hier einen regulären Ausdruck für die Verzeichnisse ein, die von der Link-Konsistenzprüfung nicht kontrolliert werden sollen. Verzeichnisse, deren Name auf das Muster passt, werden ausgelassen.

```
EXCLUDE_DIR = "test"
```

EXCLUDE_FULL_DIR

Mit dieser Direktive können Sie vollständige Pfade angeben, die von der Link-Konsistenzprüfung ausgelassen werden sollen. Nur Pfade, die exakt dem eingetragenen Wert entsprechen, werden übersprungen.

```
EXCLUDE_FULL_DIR = "/site"
```

AUTHBASIC_login

Wenn Sie das Zielsystem ganz oder teilweise mit einer Basic-Authentifizierung abgesichert haben, müssen Sie in der Konfiguration der Link-Konsistenzprüfung Authentifizierungsdaten angeben. Diese Daten nutzt die Link-Konsistenzprüfung bei der Kontrolle der Links zur Anmeldung bei dem betreffenden System. Wenn Sie diese Angaben nicht in der Konfigurationsdatei eintragen, werden sämtliche Verweise in den zugangsgeschützten Bereich im Prüfbericht als defekte Links aufgeführt. In der Spalte mit dem HTTP-Fehler erscheint dann die Meldung "401 Authorisation required".

```
AUTHBASIC_login = "login"
```

AUTHBASIC_password

Tragen Sie hier das Passwort für die Anmeldung beim zugriffsgeschützten Bereich des zu prüfenden Systems ein.

```
AUTHBASIC_password = "password"
```

ROOT_PAGES

Mit dieser Direktive definieren Sie die Startseiten für die Link-Konsistenzprüfung. Von den hier angegebenen Seiten aus folgt die Link-Konsistenzprüfung allen Verweisen. Diese Einstellung ist relevant für die Kategorie "Verwaiste Objekte" im Prüfbericht.

Mehrere Seiten notieren Sie durch Leerzeichen voneinander getrennt hintereinander.

```
ROOT_PAGES = "/index.html.de /index.html.en"
```

HTTP_PROXY

Wenn der Zugriff auf das zu prüfende System über einen Proxy-Server erfolgt, müssen Sie dessen URL in dieser Direktive eintragen.

```
HTTP_PROXY = "http://proxy:3128"
```

Wichtig:

Die Link-Konsistenzprüfung wertet die folgenden Direktiven nur aus, wenn Sie auch einen HTTP-Proxy definiert haben.

HTTPS_PROXY

Notieren Sie hier den Proxy für verschlüsselte Datenübermittlung mit SSL.

```
HTTPS_PROXY = "http://proxy:3128"
```

FTP_PROXY

Mit dieser Direktive definieren Sie den Proxy für das FTP-Protokoll.

```
FTP_PROXY = "http://proxy:3128"
```

NO_PROXY

Hier geben Sie die URLs an, auf die direkt zugegriffen werden soll.

```
NO_PROXY = "http://www.mynoproxy.de"
```

2.10.1 Konfigurationsbeispiel

Im Verzeichnis site/config Ihrer **imperia** -Installation finden Sie die Datei linkmanager.conf.sample. Kopieren Sie diese Datei nach linkmanager.conf und editieren Sie die Kopie nach Ihren Anforderungen, um die Link-Konsistenzprüfung schnell einsetzen zu können.

```
### Linkmanager Sample Configuration ###
<SERVER mycool.server>
SERVERNAME = "Livesystem"
SERVER_URL = "http://mycool.server.net"

#SETTYPE_HTML = "\..?htm.?|\..?php"
#SETTYPE_CSS = "\.css/i"


# a regexp, paths containing the given pattern are excluded
#EXCLUDE_DIR = "test"
#EXCLUDE_DIR = "backup"

# same as EXCLUDE_DIR for files
EXCLUDE_FILE = "^print"




# full path to be excluded
EXCLUDE_FULL_DIR = "/site"
EXCLUDE_FULL_DIR = "/cgi-bin"
EXCLUDE_FULL_DIR = "/icon"
EXCLUDE_FULL_DIR = "/imperia"


# define the pages from which the link consistency check is started
ROOT_PAGES = "/index.html /index.html.de /index.html.en"

# setting http_proxy directive is mandatory for the rest of the
# proxy-settings to be considered
#HTTP_PROXY = "http://www.proxy.net:8020"
HTTPS_PROXY = "http://www.proxy.net:8021"
FTP_PROXY = "http://www.proxy.net:8022"
NO_PROXY =" http://www.proxy.net:8023"

</SERVER>
```

2.11 Erweiterung des Schreibtischs

Die Einstellungen des Schreibtischs von **imperia** lassen sich nicht nur auf der Benutzerebene individuell beeinflussen (siehe Kapitel "Schreibtisch" im Benutzerhandbuch).

Als Administrator können Sie beliebige Schreibtischansichten definieren, die systemweit für alle Benutzer gültig sind. Dabei können Sie sowohl neue Funktionen hinzufügen als auch Standardfunktionen ausblenden, die nicht verfügbar sein sollen.

Diese Änderungen nehmen Sie über Perl-Module vor, die updatesicher im System gespeichert werden.

2.11.1 Schreibtischansichten einrichten

Für die Definition eigener updatesicherer Schreibtischansichten benötigen Sie folgende Komponenten:

Spaltendefinition

Die Konfiguration der verfügbaren Spalten nehmen Sie in einem Perl-Modul vor. Die Eigenschaften der einzelnen Spalten werden darin in Form eines Perl-Hashs festgehalten. Dazu gehört unter anderem die Angabe, welches Perl-Modul die Funktion der Schreibtisch-Spalte enthält.

Als Beispiel für die verfügbaren Konfigurationsmöglichkeiten können Sie sich auf Ihrem Produktionssystem die Datei `site/modules/core/Imperia/Desktop/DefaultColumns.pm` ansehen. Sie enthält die Konfiguration für die Standard-Schreibtisch-Spalten von **imperia** .

Wenn Sie eigene Spalten hinzufügen möchten, brauchen Sie nur ein Modul mit einem Hash, der die Eigenschaften der Zusatzspalten enthält, nach diesem Muster anzulegen. Es ist auch möglich, parallel mehrere Module zu nutzen.

Wichtig:

Achten Sie bei der Verwendung mehrerer Module mit Spaltendefinitionen darauf, dass Sie eindeutige Namen für Ihre zusätzlichen Spalten vergeben.

Stellen Sie sicher, dass von Ihnen verwendete Spaltennamen nicht schon für **imperia** *-Standard-Schreibtisch-Spalten in Gebrauch sind. Die von Ihnen definierten Spalten ersetzen in diesem Fall die Standard-Spalten.*

Die Dateien müssen auf Ihrem Produktionssystem in einem Verzeichnis unterhalb von `site/modules/core/` abgelegt werden.

Auf dem Schreibtisch sichtbar werden auf diese Weise definierte Spalten erst, wenn sie in eine Schreibtisch-Ansicht eingebunden werden. Zusätzlich müssen Sie durch zwei System-Konfigurationsvariablen die Erweiterung Ihres Schreibtischs aktivieren. Lesen Sie hierzu Abschnitt 2.11.2 **Schreibtisch-Erweiterungen in das System einbinden** auf Seite 34.

Im Dialog Schreibtisch-Konfiguration erscheinen von Ihnen definierte Schreibtisch-Spalten ebenfalls. Sie sind in der Pull-Down-Liste zur Auswahl hinzuzufügender Spalten aufgeführt und können auf diese Weise auch in bestehende Schreibtischansichten integriert werden.

Definition einer Schreibtisch-Ansicht

Ähnlich wie die Definition zusätzlicher Schreibtischspalten geschieht auch die Einrichtung von Schreibtischansichten durch Perl-Module. Hier werden die Eigenschaften in Form eines Perl-Hashs abgelegt. Die Definition der Standard-Schreibtischansichten liegt beispielsweise auf dem Produktionssystem in der Datei `site/modules/core/Imperia/Desktop/DefaultConfig.pm`.

Legen Sie für Ihre zusätzliche Ansicht ein eigenes Modul an. Beliebig viele Module sind möglich. Sie können aber auch alle selbst entworfenen Ansichten in einem einzigen Modul speichern.

Wichtig:

Verwenden Sie für Ihre Schreibtischansichten eindeutige Namen und beachten Sie, dass diese nicht bereits vergeben sind.

Ebenso wie Spaltendefinitionen müssen Schreibtischansichten in einem Verzeichnis unterhalb von `site/modules/core` auf dem Produktionssystem abgelegt werden. Sobald Sie ein solches Modul angelegt haben, sind die darin definierten Schreibtischansichten allen Benutzern zugänglich.

Module für Schreibtischfunktionen

Für die eigentliche Funktionalität einer Schreibtischerweiterung brauchen Sie ein separates Perl-Modul. Legen Sie es auf dem Produktionssystem im Verzeichnis `site/modules/core/Dynamic/DesktopColumn` ab.

2.11.2 Schreibtisch-Erweiterungen in das System einbinden

Zur Aktivierung eigener Schreibtischerweiterungen müssen Sie diese mit zwei Konfigurationsvariablen ins System einbinden. Die Variablen werden in der Datei `system.conf` definiert:

DESKTOP_COLUMN_MODULES

Weisen Sie dieser Variablen als Wert eine durch Semikolons getrennte Liste von den Package-Namen der Module mit Spaltendefinitionen zu.

Dazu ein Beispiel:

```
"DESKTOP_COLUMN_MODULES" = "Imperia::Desktop::myDefaultColumns;mycols::mySpecialColumns"
```

Hier werden zwei Module ins System eingebunden. Die zugehörigen Dateien sind auf dem System unter `site/modules/core/Imperia/Desktop/myDefaultColumns.pm` und `site/modules/core/mycols/mySpecialColumns.pm` abgelegt.

DESKTOP_CONFIG_MODULES

In dieser Variablen speichern Sie die Package-Namen der Perl-Module mit Ihren Schreibtischansichten. Die einzelnen Namen sind durch Semikolons getrennt zu notieren. Notieren Sie beispielsweise folgendes, um die beiden Module `site/modules/core/Imperia/Desktop/myDefaultConfig.pm` und `site/modules/core/mycols/mySpecialConfig.pm` einzubinden:

```
"DESKTOP_CONFIG_MODULES" = "Imperia::Desktop::myDefaultConfig;myconf::mySpecialConfig"
```

2.11.3 Standard-Schreibtischansichten umgehen

Der Aufbau des **imperia** -Schreibtisches findet in mehreren Phasen statt. Es werden systemweite Standardeinstellungen sowie Erweiterungen berücksichtigt. Die jeweiligen Einstellungen für Spalten und Schreibtischansichten werden dabei in einer festen Reihenfolge ausgewertet:

Reihenfolge bei der Auswertung von Spaltendefinitionen:

1. Standard-Schreibtisch-Spalten (Modul *Imperia::Desktop::DefaultColumns*).
2. Dateien mit der Endung **.col*, die unter `site/config/desktop` abgespeichert sind. Hierbei handelt es sich um Spaltendefinitionen, die von einem Benutzer über die **imperia** -Oberfläche angelegt und entsprechend kopiert wurden, damit Sie allen Nutzern des Systems zur Verfügung stehen (siehe Kapitel "Schreibtisch" im Benutzerhandbuch).
3. Spaltendefinitionen, die in der Konfigurationsvariablen *DESKTOP_COLUMN_MODULES* aufgeführt sind.
4. Benutzerspezifische Einstellungen.

Reihenfolge bei der Auswertung von Schreibtischansichten:

1. Standard-Ansichten (Modul *Imperia::Desktop::DefaultConfig*).
2. Erweiterungen, die in der Variablen *DESKTOP_CONFIG_MODULES eingetragen sind.*
3. Benutzerspezifische Einstellungen.

Sowohl bei Spaltendefinitionen als auch bei Schreibtischansichten gilt, dass später ausgewertete Einstellungen etwaige bereits bestehende überschreiben.

Auf diese Weise können Sie beispielsweise die Anzeige von bestimmten Spalten auch in den Standard-Ansichten unterdrücken. Diese Änderung gilt dann für alle Benutzer.

Definieren Sie dazu eine eigene Schreibtisch-Ansicht mit dem Namen der betreffenden Standard-Ansicht und lassen Sie die Spalte aus, die Sie ausblenden wollen. Dazu ein Beispiel:

Sie möchten die Anzeige des Dokumententitels in der Default-Ansicht des Schreibtisches unterbinden. Die Definition der Ansicht steht im Modul `site/modules/core/Imperia/Desktop/DefaultConfig.pm`:

```
# Ausschnitt aus dem Package Imperia::Desktop::DefaultConfig

Default => {
                columns => [
                                        [ qw (grid_status) ],
                                        [ qw (edit) ],
                                        [ qw (title) ],
                                        [ qw (category) ],
                                        [ qw (modified) ],
                                        [ qw (next_step) ],
                                        ],
                        sort => [ qw (modified created node_id) ],
                },
```

Natürlich können Sie auf die selbe Weise auch Erweiterungen der Standard-Ansichten vornehmen.

Um die Spalte auszublenden, die den Dokumententitel anzeigt, legen Sie ein eigenes Modul an und definieren darin die Default-Ansicht entsprechend:

```
# Ausschnitt aus dem Package Imperia::Desktop::myDefaultConfig

Default => {
                columns => [
                                        [ qw (grid_status) ],
                                        [ qw (edit) ],
                                        [ qw (category) ],
                                        [ qw (modified) ],
                                        [ qw (next_step) ],
                                        ],
                        sort => [ qw (modified created node_id) ],
                },
```

2.12 Skinning der Flexmodul-Steuerelemente

Den Quellcode der Flexmodul-Steuerelemente können Sie leicht abändern oder komplett ersetzen, indem Sie den **imperia** Standard-Code durch selbst erstellte Skins ersetzen.

Flexmodul-Skins ähneln grundsätzlich Codeincludes. Legen Sie jeweils für die Flexmodul-Kontrollleiste und die Steuerelemente der Flexinstanzen eine eigene Datei mit dem jeweiligen Quellcode an und speichern Sie diese im Verzeichnis `site/skin/flex`, in dem Sie auch Beispieldateien für Flexmodul-Skins finden. Die Namenskonvention für Skindateien lautet wie folgt:

```
MYNAME.control.htms => Skin-Datei für die Flexmodul-Kontrollleiste
MYNAME.instance.htms => Skin-Datei für die Steuerelemente der Flexinstanzen
```

Wichtig:

MYNAME muss für beide Dateien gleich lauten, damit die Einbindung über den Flexmodulaufruf im Template funktioniert.

Die Einbindung Ihrer Flexmodul-Skins können Sie entweder dokumentenbasiert durch einen Parameter im Flexmodulaufruf oder durch zwei Variablen zur Systemkonfiguration vornehmen. Für Einzelheiten zum Flexmodul-Skinning auf Dokumentenbasis lesen Sie bitte den Abschnitt **SKIN** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

2.12.1 FLEXMODULE-SKIN

Mit dieser Variable geben Sie den Dateinamen für Ihr Flexmodul-Skin an, mit dem Sie systemweit das tabellenbasierte Standard-Layout der Flexmodul-Steuerleiste überschreiben möchten. Geben Sie nur den Dateinamen an, wenn Sie die Datei im Standard-Verzeichnis für Flexmodul-Skins, `site/skins/flex` ablegen möchten. Alternativ können Sie eine Pfadangabe ausgehend von der Document-Root Ihres Entwicklungssystems notieren.

2.12.2 FLEXINSTANCE-SKIN

Wie `FLEXMODULE-SKIN`, nur für die Steuerleiste der Flex-Instanzen.

2.13 Änderungsverfolgung

Die administrativen Einstellungen für die Änderungsverfolgung beschränken sich auf die Bestimmung der Farben, mit denen die Änderungen dargestellt werden. Diese Farben werden in einem gesonderten Stylesheet definiert.

Folgende Änderungen können angezeigt werden:

- hinzugefügter Text (im Standard grün und unterstrichen dargestellt).
- gelöschter Text (im Standard rot und durchgestrichen dargestellt).
- hinzugefügte Tabellen, Tabellenzellen und Anker (im Standard hellgrün dargestellt).
- gelöschte Tabellen, Tabellenzellen und Anker (im Standard orange dargestellt).
- geänderte Bilder (im Standard mit einem durchgehenden orangefarbenen Rand dargestellt).
- eingefügte Zeilenumbrüche (im Standard wird der umbrochene Text in der neuen Zeile grau dargestellt).
- Verweise, auf den ein IFrame zeigt (im Standard wird der IFrame mit einem durchgehenden orangefarbenen Rand dargestellt).

Es bestehen zwei Möglichkeiten, die Farben zu ändern, die für die Darstellung der Änderungen verwendet werden:

- für das gesamte System, in dem die Standardeinstellungen im entsprechenden Stylesheet geändert werden.
- über Style-Angaben im Template.

2.13.1 Änderung der Standardfarben

Werden die Standardfarben der Änderungsverfolgung im Stylesheet geändert, gelten diese Änderungen für alle Dokumente im gesamten Produktionssystem.

Das für die Farben der Änderungsverfolgung verantwortliche Stylesheet liegt im Verzeichnis `<Document-Root des Produktionssystems>/imperia/skin/html/css` unter dem Namen `diff.css`.

Es stehen folgende CSS-Formate zur Verfügung:

imperiaPlusSpan

Bestimmt das Aussehen von hinzugefügtem Text.

imperiaMinusSpan

Bestimmt das Aussehen von gelöschtem Text.

imperiaPlusSpan table, td, a

Bestimmt das Aussehen von hinzugefügten Tabellen, Tabellenzellen und Ankern.

imperiaMinusSpan table, td, a

Bestimmt das Aussehen von gelöschten Tabellen, Tabellenzellen und Ankern.

imperiaPlusSpan img iframe

Bestimmt das Aussehen von hinzugefügten Bildern und IFrames.

imperiaMinusSpan img iframe

Bestimmt das Aussehen von gelöschten Bildern und IFrames.

imperiaNewlineChangesSpan

Bestimmt das Aussehen von umbrochenem Text.

imperiaImageChanged

Bestimmt das Aussehen von geänderten Bildern.

imperiaIFrameChanged

Bestimmt das Aussehen von geänderten IFrames.

Eine weitere Möglichkeit, die Standardfarben zu ändern, besteht darin, innerhalb des Templates Meta-Variablen zu erzeugen, die andere Farbwerte enthalten. Lesen Sie hierzu das Kapitel **Änderungsverfolgung** im Programmierhandbuch.

2.14 Reparsen umfangreicher Sites

Zum Reparsen einer umfangreichen Website steht das Skript `site/bin/xreparse.pl` zur Verfügung. Eine Durchführung über die **imperia** -Oberfläche ist nicht vorgesehen. Dieses Skript ist im Gegensatz zu der entsprechenden über die Benutzeroberfläche ausführbaren Funktionen auch für große Datenmengen geeignet.

Mit Hilfe des Skripts können alle Dokumente, die sich in der Freischaltliste oder auf dem Zielsystem befinden, bearbeitet werden. Es wird vorausgesetzt, dass sich keines dieser Dokumente im Workflow befindet. Wir empfehlen weiterhin, den Workflow für alle Dokumente zu beenden, bevor dieses Skript ausgeführt wird.

xreparse.pl ignoriert Dokumente des Media Asset Management, es sei denn, dies wird mit dem Parameter -m erzwungen.

Hinweis:

Die Meta-Variable `publish_date` *wird in den Dokumenten, die auf den mit dem Parameter* `-p` *gesetzten Veröffentlichungsbedingungen passen, auf den Wert* `2001-01-01 01:01` *gesetzt.*

2.14.1 Optionen

Das Skript wird auf der Kommandozeile mit folgenden Optionen gesteuert:

-h

Zeigt eine kurze Hilfe zum Skript.

-k

Behält das Veröffentlichungsdatum beim Reparsen bei ("keep").

-p *Argument*

Dieser Parameter kontrolliert das Veröffentlichungsverhalten der reparsten Dokumente. Als Argumente sind `none` und `all` erlaubt.

Wird das Argument `none` gesetzt, werden alle Dokumente nach dem Reparsen in die Freischaltliste eingetragen, werden aber nicht freigeschaltet.

Wird das Argument `all` gesetzt, werden alle Dokumente, die sich bereits auf dem Zielsystem befinden, nach dem Reparsen erneut freigeschaltet, wobei auch andere Dokumente, die sich in der Freischaltliste befinden, mit freigeschaltet werden.

Wird die Option **-p** nicht verwendet, werden alle reparsten Dokumente, die sich bereits auf dem Zielsystem aber nicht in der Freischaltliste befinden, wieder freigeschaltet.

-l

Dieser Parameter reparst Dokumente und fügt sie anschließend der Freischaltliste hinzu oder veröffentlicht die Dokumente, wenn sie sich schon auf dem Zielsystem befinden.

-w

Die Dokumente werden nicht reparst, sondern re-importiert und in den Workflow geleitet. Zusätzlich setzt dieser Parameter die Umgebungsvariable `__IMPERIA_XREPARSE` auf einen wahren Wert. Die Variable können Sie dazu verwenden, reparste Dokumente im Workflow mit Hilfe eines IfElse-Plug-Ins in einen anderen Workflow-Zweig zu lenken. Die Bedingung dieses IfElse-Plug-Ins sollte wie folgt aussehen:

```
"<!--ENV:__IMPERIA_XREPARSE-->" AND (NOT "<!--ENV:GATEWAY_INTERFACE-->")
```

-f

Diese Option erzwingt die Ausführung des Skripts ohne weitere Benutzereingabe. Wird die Option nicht verwendet, wird eine Empfehlung, `STDOUT` in eine Log-Datei umzulenken, angezeigt.

-e

Wird diese Option in Zusammenhang mit **-w** verwendet, wird der erste Workflow-Schritt eines jeden reparsten Dokuments ausgeführt. Das Standard-Verhalten ist, den ersten Workflow-Schritt zu überspringen.

-n

Wird diese Option verwendet, interpretiert das Skript eine eventuell vorhandene Liste von Argumenten als Liste von NodeIDs, während standardmäßig eine Liste von Argumenten als Liste von Template-Namen interpretiert wird. Dadurch ist es möglich, das Reparsen auf eine Rubrik und deren Unterrubriken zu beschränken.

-m

Diese Option erzwingt das Reparsen auch für die Dokumente der MAM. Da deren Reparsen selten sinnvoll ist, werden MAM-Dokumente im Allgemeinen von xreparse.pl nicht bearbeitet.

2.14.2 Argumente

Standardmäßig interpretiert das Skript eine Liste von Argumenten als Liste von Template-Namen, wobei das führende *template* und das *htms*-Suffix nicht angegeben werden. Wird die Option **-n** verwendet, wird die Liste der Argumente jedoch als Liste von Rubriken-NodeIDs interpretiert.

Das Reparsen wird auf die Dokumente beschränkt, deren Templates auf die angegebenen Namen passen bzw. deren NodeIDs mit einer der angegebenen Rubriken-NodeIDs beginnen.

2.14.3 Beispiel

Dieses Beispiel eignet sich für einen Workflow, dessen erster Schritt interaktiv, alle folgenden Schritte jedoch nicht interaktiv sind. Dadurch können die reparsten Dokumente den Workflow durchlaufen, ohne dass ein Benutzereingriff notwendig ist.

```
./xreparse.pl -w -p all -f -e Service Produkt
```

Es werden alle Dokumente reparst, die mit dem Template `templateService.htms` oder dem Template `templateProdukt.htms` erzeugt wurden. Beim Reparsen wird die Umgebungsvariable gesetzt. Es bedarf keiner weiteren Benutzereingabe. Nach dem Reparsen wird der erste Workflow-Schritt ausgeführt. Die reparsten Dokumente werden freigeschaltet.

2.15 Das interne Mailsystem

Auf das **imperia** -eigene Mailsystem kann über das Briefsymbol in der Kopfzeile zugegriffen werden. Es bedarf keiner Administration. Es kann aber deaktiviert werden, indem über die Verwaltung der Rechte auf die Module die Zugriffsrechte auf den Controller `/system/mailer` entfernt werden.

Hinweis:

Für Benutzer mit Superuser-Rechten hat diese Einstellung keinen Effekt, da Superuser immer alle Rechte haben.

Die Funktionsbeschreibung lesen Sie bitte im Kapitel "Das interne Mailsystem" im Benutzerhandbuch nach.

Kapitel 3. Struktur

Unter dem Menüpunkt **Struktur** finden Sie Funktionen zur Verwaltung von Rubriken, Metadateien, Templates, Flexmodulen, Codeincludes, Slots-Modulen, Verschieben von Assets und den Template Reparser. Informationen zum Erstellen bzw. Programmieren dieser Dateien finden jeweils in einem eigenen Kapitel. In diesem Kapitel sind lediglich die Verwaltung und Handhabung dieser speziellen Dateien über die **imperia** -Oberfläche beschrieben.

Weitere Themen sind Dynamische Module, der Template-Reparser und der Abgleich der Systemdateien zwischen Produktions- und Zielsystem.

3.1 Rubriken

Das Rubriken-Management erreichen Sie durch Auswahl des Menüpunktes **Menü** -> **Struktur** -> **Rubriken**. Sie gelangen zu folgendem Dialog (hier mit Beispielrubriken):

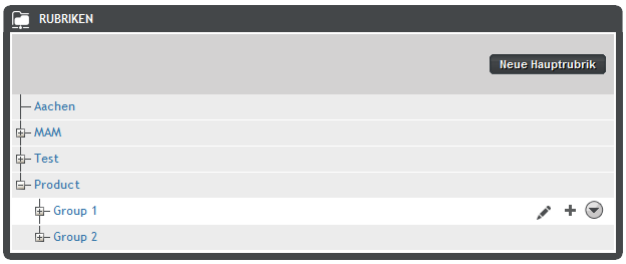


Abb. 3.1: Dialog Rubriken bearbeiten

Hier finden Sie Funktionen zum Einrichten, Bearbeiten und Löschen von Rubriken und zum Bearbeiten der jeweiligen Zugriffsrechte. Bestehende Rubriken werden in einer Baumstruktur angezeigt.

Rubriken müssen nicht zwangsläufig Dokumente enthalten. Sie können durch Schachtelung auch die thematische Struktur Ihres Projektes abbilden. Einige Rubriken enthalten dabei dann lediglich weitere Unterrubriken. Lesen Sie hierzu auch Abschnitt 5.1.1 **Rubriken-Informationen angeben** auf Seite 70.

Die Einstellung, ob in einer Rubrik Dokumente angelegt werden können oder nicht, kann jederzeit geändert werden. Dadurch kann man eine Rubrik "einfrieren" und vermeiden, dass weitere Dokumente angelegt werden.

Der hier erzeugte Rubrikenbaum wird auch im Archiv, beim Erstellen neuer Dokumente und beim Bearbeiten der Rubriken verwendet. Für jede dieser Ansichten wird jedoch getrennt abgespeichert, welche Rubriken aufgeklappt sind und welche nicht.

Das Rubriken-Management steht nur Rollen mit entsprechender Zugriffsberechtigung zur Verfügung. Wenn Sie den Menüpunkt **Rubriken** nicht im Menü sehen, hat Ihre derzeitige Rolle keinen Zugriff auf diese Funktion. Nehmen Sie eine Rolle mit ausreichender Berechtigung an oder wenden Sie sich an den **imperia** -Systemadministrator (**Superuser**).

3.1.1 Neue Rubrik anlegen

Neue Rubriken werden entweder als Hauptrubrik, also als ein Wurzelelement des Rubrikenbaums, oder als Unterrubrik einer Hauptrubrik angelegt. Eine neue Hauptrubrik wird über den Link **Neue Hauptrubrik** in der Titelleiste des Dialogs erzeugt.

Wenn Sie eine Unterrubrik anlegen wollen, klicken Sie in der Zeile der entsprechenden Hauptrubrik auf folgenden Button:

Halten Sie dabei gleichzeitig die Taste *STRG* gedrückt, um alle Unterknoten zu öffnen bzw. zu schließen. Neue Unterrubriken werden immer unterhalb der Zeile der Hauptrubrik angelegt. Nach dem Betätigen des Buttons wird der Dialog **Neue Rubrik anlegen** geöffnet.

Dieser Dialog ist in vier Abschnitte unterteilt, die in Kapitel 5.1 beschrieben werden. Eine genaue Beschreibung der einzelnen Abschnitte finden Sie in Abschnitt 5.1 **Rubriken-Eigenschaften bestimmen** auf Seite 69.

Eingaben werden über den Button **Okay** am unteren Rand des Dialogs gespeichert.

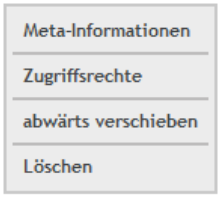
3.1.2 Rubriken bearbeiten

In der Spalte **Bearbeiten** des Dialogs **Rubriken bearbeiten** befindet sich in jeder Zeile ein Button, mit dem die entsprechende Rubrik zum Bearbeiten geöffnet werden kann:

Der Button öffnet den Dialog **Rubrik bearbeiten**, in dem Sie die vorhandenen Einstellungen in den entsprechenden Abschnitten ändern können. Zur Speicherung der Änderungen klicken Sie auf den Button **Speichern** am unteren Rand der **imperia** -Oberfläche.

3.1.2.1 Rubriken-Metainformationen

Alle zusätzlich Einstellungen zu den Kategorien können über das Submenü angsprochen werden.



Neben den Einstellungen, die Sie bei der Erstellung der Rubriken vornehmen, können Sie bei der Bearbeitung bereits bestehender Rubriken Metainformationen mit der Rubrik verknüpfen. Klicken Sie hierzu auf den Link **Bearbeiten**. Sie gelangen zu einem Meta-Viewer, in dem Sie Metainformationen hinzufügen, entfernen oder ändern können.

Hinweis:

Diese Funktion steht nur Benutzern mit Schreibrecht auf die Rubrik zur Verfügung.

Alternativ können Sie den Dialog zum Bearbeiten der Rubrik-Metainformationen auch direkt aus der Rubriken-Verwaltung aufrufen. Klicken Sie hierzu auf den Eintrag **Meta-Informationen** im Zeilenmü der Rubrikenverwaltung.

Rubrik-Metainformationen vererben sich automatisch beim Anlegen von Unterrubriken. Das heißt, die Metainformationen sind dort dann mit den Metainformationen der übergeordneten Rubrik vorbelegt.

Hinweis:

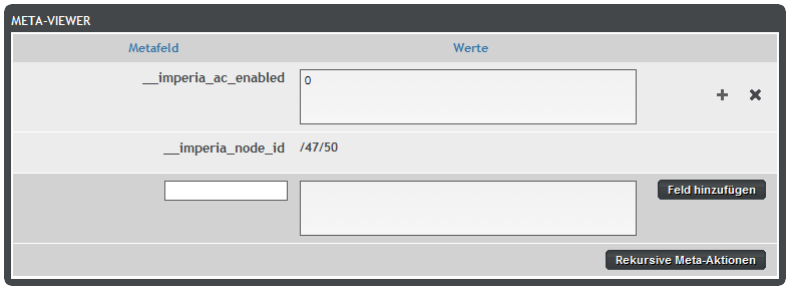
Der Zugriff auf Rubrik-Metainformationen in Dokumenten lässt sich etwas einfacher mit dem Workflow-Plug-In Cat2DocMeta umsetzen. Lesen Sie hierzu ???.

Wozu brauche ich diese Meta-Informationen? Statt das DESCRIPTION-Feld der Rubriken mit Werten zu füllen, um zum Beispiel Variablen für die Navigationserzeugung zu speichern, kann man hier sehr umfangreich eigene Variablen an Rubriken hängen. Auf diese kann man über Perl sehr einfach zugreifen: wo immer der Nodeinfo-Hash der Rubrik zurückkommt oder gesetzt wird, gibt es unter dem Key `META_INFO` ein Meta::Info-Objekt. Die Klasse Category::Category wurde um *getMetaInfo()* und *setMetaInfo()* erweitert. Wer bei Verwendung der Filesystem-Datenhaltung die Meta-Informationen an **imperia** vorbei mit stdump/stfreeze lesen bzw. manipulieren will, sollte beachten, dass dort nicht das Meta::Info-Objekt, sondern die blanke Hash-Referenz, bei der es sich um kein Objekt handelt, abgespeichert wird. In der Datenbank dagegen liegen die Meta-Informationen in der neuen Tabelle `category_meta`.

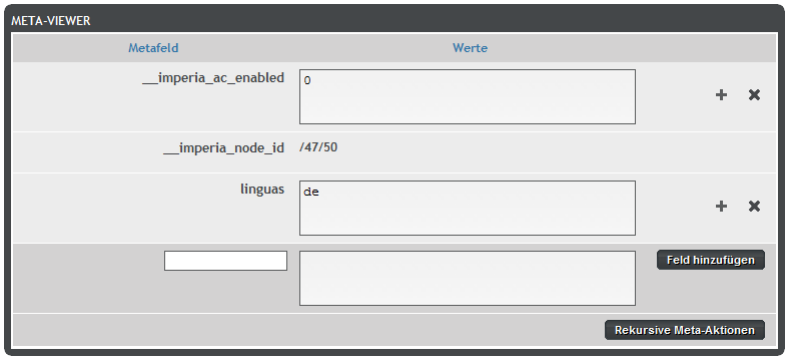
3.1.2.2 Rubriken-Metainformationen bearbeiten

Enthält das Metafeld *__imperia_category_template* einer Rubrik einen gültigen Templatenamen, wird zum Editieren stattdessen dieses Template verwendet. Falls das Metafeld nicht vorhanden ist, wird die System-Conf-Variable *CATEGORY_TEMPLATE* herangezogen und ansonsten das Default-Template verwendet. Mit diesem Mechanismus können Sie also die Verwendung ganz bestimmter Metainformationen erreichen.

Im Folgenden wird die Verwendung des **imperia** -Fallback-Templates erklärt.



Um eine Metainformation einzugeben, tragen Sie den Namen des neuen Feldes in das linke, kleinere Feld ein und den Wert in das rechte, größere Feld. In unserem Beispiel ist das „linguas“ und „de“. Klicken Sie dann auf die Schaltfläche "<Feld hinzufügen>". Wenn Sie mehrere Werte als Array speichern wollen, klicken Sie auf das Plus-Icon und ein neues leeres Eingabefeld erscheint.



3.1.2.3 Zugriff auf die Rubriken-Metainformationen

Der Zugriff auf die Metainformationen erfolgt wie bei jedem anderen Rubrikenfeld. Ein eventueller *[index]* in Klammern liest aus einem Array das entsprechende Feld aus.

Alternativ können Sie einem Meta-Feld auch *META_INFO_* voranstellen, um normale Rubrikenfelder von Metafeldern zu unterscheiden. Das ist dann nützlich, wenn Sie einen Namen verwenden möchten, der bereits als normales Feld in Rubriken verwendet wird. Das wird selten notwendig sein, kann aber in Ausnahmefällen helfen.

Dazu ein Beispiel:

Angenommen, die Meta-Informationen der Rubrik mit Namen "Test", der ID "/123/456/789" und dem Directory "/test" sind folgendermaßen befüllt:

```
example | "Ein Beispieltext"
DIRECTORY | "Meta-Info, nicht das echte Directory"
list | "eins" | "zwei" | "drei"
```

In einem (Dokumenten!)-Template verwenden Sie jetzt:

```
category: <!--SECTION:NAME:0--><br />
example(M): <!--SECTION:META_INFO_example:0--><br />
example: <!--SECTION:example:0--><br />
DIRECTORY: <!--SECTION:DIRECTORY:0--><br />
DIRECTORY(M): <!--SECTION:META_INFO_DIRECTORY:0--><br />
List: <!--SECTION:list[0]:0--><br />
List: <!--SECTION:list[1]:0--><br />
List: <!--SECTION:list[2]:0--><br />
List: <!--SECTION:list:0--><br />
Rubriken-ID: <!--SECTION:__imperia_node_id:0--><br />
```

Dann macht der Template-Prozessor daraus:

```
category: Test
example(M): Ein Beispieltext
example: Ein Beispieltext
DIRECTORY: /test
DIRECTORY(M): Meta-Info, nicht das echte Directory
List: eins
List: zwei
List: drei
List: eins
Rubriken-ID: /123/456/789
```

Beispiel 3.1. Zugriff auf Meta-Informationen von Rubriken

3.1.2.4 Verfügbare Sprachen für mehrsprachige Dokumente festlegen

Ein weiteres Beispiel für die Verwendung von Rubrik-Metainformationen ist das Metafeld `linguas`. Es speichert die gewünschten Sprachen in einem Array. Jedes Element dieses Arrays besteht aus einem ISO 639 Sprach-Code für eine Sprache (de, en, fr etc.). **imperia** baut aus dem Inhalt dieser Metavariablen die Sprachauswahl im Meta-Edit und im Bearbeiten-Schritt des Workflows auf.

3.1.3 Rubrikenrechte einstellen

Für jede Rubrik können für jede Rolle im System verschiedene Zugriffsrechte eingestellt werden. Diese Rechte bestimmen die Funktionen, die einem Benutzer für die Bearbeitung von Dokumenten in dieser Rubrik zur Verfügung stehen. Mögliche Rechte sind:

- Lesen
- Dokumente erzeugen
- Schreiben
- Löschen

Die Rechte einer Rolle in einer Rubrik lassen sich über die Funktion **Rubrik bearbeiten** (siehe Abschnitt 3.1.2 **Rubriken bearbeiten** auf Seite 41) und im Dialog **Rubriken-ACL-Verwaltung** einstellen. Diesen Dialog öffnen Sie für jede Rubrik über den Button **Zugriffsrechte** im Dialog **Rubriken bearbeiten**. Einer der Hauptunterschiede zu der im vorigen Abschnitt beschriebenen Bearbeitung der Rechte ist die Möglichkeit, in der ACL-Verwaltung Zugriffsrechte rekursiv einzustellen. Sie können hier also gleich für ganze Teilrubrikenbäume die Berechtigungen einstellen.

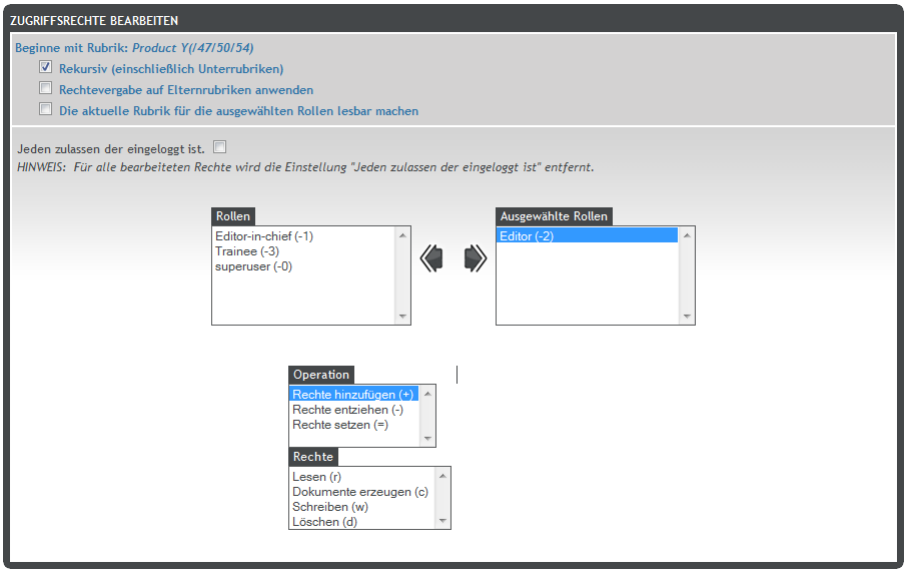


Abb. 3.2: Rubriken-ACL-Verwaltung

Der Dialog ist in zwei Abschnitte unterteilt:

Im ersten Abschnitt **Parameter** legen Sie fest, ob die Änderung der Rechte rekursiv, also auch für enthaltene Unterrubriken, gelten soll.

Im zweiten Abschnitt **Rollen hinzufügen** weisen Sie die Rechte einzelnen oder mehreren Rollen zu. Die Liste **Rollen** enthält alle verfügbaren Rollen. Markieren Sie in der Liste die Rolle(n), die Sie modifizieren wollen. Für eine Mehrfachauswahl halten Sie gleichzeitig die *STRG*-Taste gedrückt.

Ihnen stehen die folgenden Optionen zur Auswahl:

Rechte hinzufügen (+)

Diese Funktion fügt die ausgewählten Rechte den bereits bestehenden Rechten der ausgewählten Rollen hinzu.

Rechte entziehen (-)

Hiermit werden die markierten Rechte den markierten Rollen entzogen.

Rechte setzen (=)

Diese Funktion setzt die Rechte der ausgewählten Rollen auf die markierten Rechte.

Wichtig:

Nicht markierte Rechte werden den Rollen entzogen.

In der Liste **Rechte** stehen die Rechte zur Auswahl, die mit der ausgewählten Aktion für die markierten Rollen modifiziert werden sollen.

Der Button **Okay** führt die eingestellten Änderungen der Zugriffsrechte aus, der Button **Schließen** schließt den Dialog, ohne Änderungen zu berücksichtigen.

3.1.4 Rubrikenreihenfolge ändern

Die Reihenfolge der Einträge im Rubrikenbaum lässt sich verändern. Änderungen wirken sich dabei auch auf etwaige Unterrubriken aus. Zur Änderung der Rubrikreihenfolge verschieben Sie die gewünschte Rubrik zeilenweise über die Einträge „aufwärts verschieben“ „abwärts verschieben“ innerhalb ihrer Ebene.

Rubriken lassen sich nur auf der Ebene verschieben, auf der Sie sie angelegt haben. Es ist nicht möglich, die Rubrikenhierarchie zu ändern. Sie können mit dieser Funktion also weder eine Hauptrubrik zur Unterrubrik einer anderen machen, noch eine Unterrubrik in eine Hauptrubrik umwandeln. Um dies zu realisieren, stehen die Adminskripte zur Verfügung.

3.1.5 Rubriken löschen

Das Löschen einer Rubrik geschieht über den Eintrag „Löschen“ im Untermenü in der entsprechenden Zeile.

Die ausgewählte Rubrik wird gelöscht.

Sie können nur Rubriken löschen, die keine Dokumente oder Unterrubriken enthalten. Versehentliches Löschen von gefüllten Rubriken wird durch eine Fehlermeldung verhindert.

3.2 Metadateien

Über das Metadatei-Management werden alle Metadateien gepflegt. Über einen Dialog können Metadateien bearbeitet, gelöscht, neu erzeugt und aus dem Dateisystem in das **imperia** -System geladen werden.

Ferner können Rechte für jede Metadatei gesetzt oder entzogen werden. Hierzu gehören die Rechte

- Schreiben,
- Ändern,
- Lesen und
- Löschen.

Zum Ändern von Dateirechten lesen Sie bitte Abschnitt 3.11 **Dateirechte in imperia verwalten** auf Seite 60.

Um das Metadatei-Management zu öffnen, wählen Sie Wählen Sie **Menü** -> **Struktur** -> **Metadatei**. Folgender Dialog öffnet sich daraufhin:

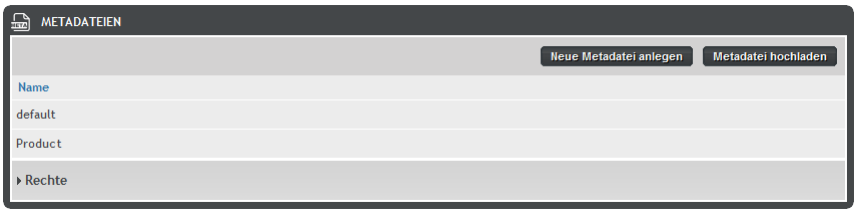


Abb. 3.3: Dialog Metadateien bearbeiten

Der Dialog ist in drei Abschnitte gegliedert:

Dateiliste

Diese Liste enthält alle eingerichteten Metadateien. Zusätzlich befinden sich hier Buttons für die Funktionen **Bearbeiten**, **Zugriffsrechte** und **Löschen**. Der Button **Bearbeiten** öffnet den internen Editor, in dem Metadateien editiert werden. Mit Hilfe des Buttons **Zugriffsrechte** wird ein Dialog geöffnet, in dem Leserechte und Rechte zum Ändern für einzelne Metadateien gesetzt werden. Über den Button **Löschen** wird die entsprechende Metadatei im System und im Dateisystem gelöscht.

Rechteverwaltung

In diesem Abschnitt, der beim Laden des Metadatei-Managements geschlossen ist und sich erst nach einem Klick auf das Pluszeichen komplett öffnet, werden Schreib-, Lese- und Löschrechte für mehrere Metadateien gesetzt. Lesen Sie hierzu Abschnitt 3.11 **Dateirechte in imperia verwalten** auf Seite 60.

Links

Unterhalb der Rechteverwaltung befinden sich die Links **Neue Metadatei anlegen** und **Metadatei hochladen**. Lesen Sie zu diesen Funktionen Abschnitt 3.2.1 **Neue Metadatei anlegen** auf Seite 45 bzw. Abschnitt 3.2.2 **Metadatei hochladen** auf Seite 46.

3.2.1 Neue Metadatei anlegen

Um eine neue Metadatei anzulegen, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 3.2 **Metadateien** auf Seite 45 beschrieben.

2. Klicken Sie auf den Link **Neue Metadatei anlegen**. Folgender Dialog öffnet sich:

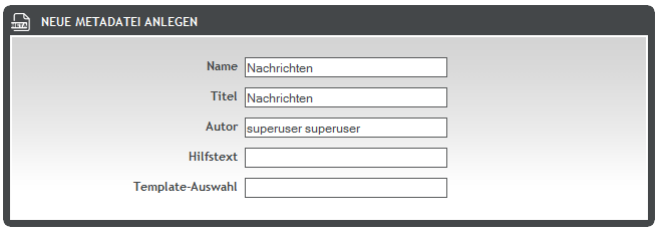


Abb. 3.4: Dialog Neue Metadatei anlegen

Tragen Sie in die einzelnen Felder die entsprechenden Daten ein. **imperia** generiert anhand dieser Daten eine neue Metadatei und legt diese im Verzeichnis `/site/metafiles` ab.

Den Hilfstext bekommen die Benutzer im Meta-Edit-Schritt angezeigt. Im Quelltext der Metadatei finden Sie ihn in der Zuweisung `HELPTEXT = "..."`.

Das Feld *Template-Auswahl* ermöglicht Ihnen, den Benutzern die Wahl zu lassen, selbst das Template aus einem Pool von Templates für ihr Dokument zu bestimmen. Tragen Sie eine kommaseparierte Liste mit den Namen der Templates, die auswählbar sein sollen, ohne Dateiendungen in das Feld ein. Im Quellcode der Metadatei finden Sie diese Liste in der Zuweisung `TEMPLATESELECT "template1, template2, ..."`. Wenn Sie keine Auswahlmöglichkeit für Templates benötigen, lassen Sie das Feld leer. Dann wird automatisch das Template aus den Rubrikeinstellungen verwendet. Weitere Information hierzu finden Sie im Abschnitt **Syntax-Referenz** im Kapitel **Metadateien** des Programmierhandbuches.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

   Die neue Datei wird in die Dateiliste eingetragen und kann von dort über den Button **Bearbeiten** geöffnet und editiert werden (siehe Abschnitt 3.2.3 **Metadatei bearbeiten** auf Seite 47).

3.2.2 Metadatei hochladen

Metadateien können Sie auch außerhalb von **imperia** mit einem beliebigen Texteditor erstellen. Eine eingehende Beschreibung der Syntax und des Aufbaus finden Sie im Kapitel **Metadateien** des Programmierhandbuchs.

Hinweis:

Beachten Sie, dass Metadateien UTF-8-codiert sein müssen (ohne BOM).

Um eine solche Datei in das System einzubinden, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 3.2 **Metadateien** auf Seite 45 beschrieben.
2. Klicken Sie unterhalb der Rechteverwaltung auf den Link **Metadatei hochladen**.

   Es wird folgender Dialog geöffnet:

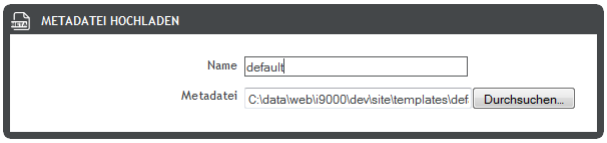


Abb. 3.5: Dialog Metadatei hochladen

3. Tragen Sie in die Felder die entsprechenden Daten ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie eine Metadatei im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis `/site/metafiles` kopiert und steht anschließend im System zur Verfügung.

3.2.3 Metadatei bearbeiten

Metadateien können im internen Editor oder einem anderen Text-Editor bearbeitet werden. Alle Metadateien sind im Verzeichnis `/site/metafiles` gespeichert. Um eine Metadatei im internen Editor zu editieren, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 3.2 **Metadateien** auf Seite 45 beschrieben.
2. Klicken Sie auf den Button **Bearbeiten** neben dem entsprechenden Metadatei-Eintrag.

   In einem weiteren Fenster wird die ausgewählte Metadatei im internen Editor geöffnet.

   Die Bedienung des internen Editors wird im Kapitel **Der interne Editor** im Autorenhandbuch beschrieben.

Tipp

Der interne Editor ist auch eine einfache Methode zur "Konvertierung" von Metadateien, die nicht im UTF-8 Zeichensatz erstellt worden sind. Kopieren Sie den Inhalt der Datei in die Zwischenablage und von dort aus in das Editorfenster. **imperia** *speichert die Datei automatisch im Unicode-Zeichensatz ab.*

3.2.4 Metadatei löschen

Um eine Metadatei zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 3.2 **Metadateien** auf Seite 45 beschrieben.
2. Klicken Sie auf den Button **Löschen** neben dem entsprechenden Eintrag in der Dateiliste.

   Es erscheint eine Sicherheitsabfrage, die Sie mit **Okay** bestätigen müssen. Anschließend wird die Metadatei aus **imperia** und aus dem Verzeichnis `/site/metafiles` gelöscht. Der Button **Zurück** bricht die Aktion ab.

3.3 Templates

Mit Hilfe des Template-Managements werden alle vorhandenen Templates gepflegt. Über einen Dialog können Templates bearbeitet, gelöscht, neu erzeugt und aus dem Dateisystem in das **imperia** -System geladen werden.

Ferner können Rechte für einzelne oder gleichzeitig für mehrere Templates gesetzt bzw. entzogen werden. Im Einzelnen sind das folgende Rechte:

- Schreiben,
- Ändern,
- Lesen und
- Löschen.

Zum Ändern von Dateirechten lesen Sie bitte Abschnitt 3.11 **Dateirechte in imperia verwalten** auf Seite 60.

Um das Template-Management zu öffnen, wählen Sie **Menü** -> **Struktur** -> **Templates**.

Sie gelangen zu folgendem Dialog:

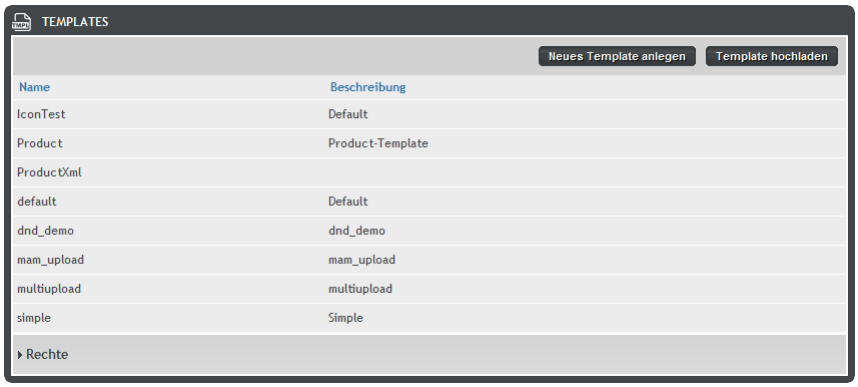


Abb. 3.6: Dialog Templates bearbeiten

Der Dialog ist in drei Abschnitte gegliedert:

Dateiliste

Diese Liste enthält die Namen (Spalte ganz links) und die Beschreibungen (zweite Spalte von links) aller bereits eingerichteten Templates. Zusätzlich erscheinen beim Überfahren der Zeile die zusätzlichen Funktionen. Ein Klick auf den Namen bzw. die Beschreibung führt zum **Bearbeiten** des Templates. Über das Kreuz kann des jeweilige Template im CMS und im Dateisystem gelöscht werden. Die Steuerung der **Rechte** kann über das Kontektmenü der Zeile aufgerufen werden. Hier wird ein Dialog geöffnet, in dem Leserechte und Rechte zum Ändern für einzelne Templates gesetzt werden.

Rechteverwaltung

In diesem Abschnitt, der beim Laden des Template-Managements geschlossen ist und sich erst nach einem Klick auf das Pluszeichen komplett öffnet, werden Schreib-, Lese- und Löschrechte für mehrere Templates gesetzt. Lesen Sie hierzu Abschnitt 3.11 **Dateirechte in imperia verwalten** auf Seite 60.

Links

Unterhalb der Rechteverwaltung befinden sich die Links **Neues Template anlegen** und **Template hochladen**. Lesen Sie zu diesen Funktionen Abschnitt 3.3.1 **Neues Template anlegen** auf Seite 48 und Abschnitt 3.3.2 **Template hochladen** auf Seite 48.

3.3.1 Neues Template anlegen

Um ein Template neu anzulegen, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 3.3 **Templates** auf Seite 47 beschrieben.
2. Klicken Sie auf den Link **Neues Template anlegen**. Es wird folgender Dialog geöffnet:

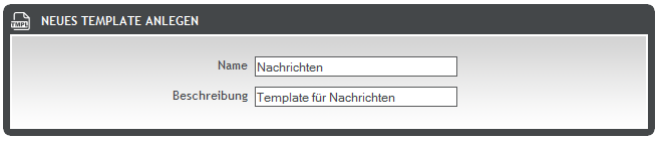


Abb. 3.7: Dialog Neues Template anlegen

   Tragen Sie in die einzelnen Felder die entsprechenden Daten ein. **imperia** generiert anhand dieser Daten eine neues Template und legt dieses im Verzeichnis `/site/templates` ab.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

   Die neue Datei wird in die Dateiliste eingetragen und kann von dort über den Button **Bearbeiten** im internen Editor geöffnet und editiert werden (siehe Abschnitt 3.3.4 **Template bearbeiten** auf Seite 49).

3.3.2 Template hochladen

Templates lassen sich mit einem beliebigen Texteditor erstellen. Führen Sie folgende Schritte aus, um ein extern erstelltes Template auf Ihr **imperia** -System hochzuladen:

1. Öffnen Sie das Template-Management wie unter Abschnitt 3.3 **Templates** auf Seite 47 beschrieben.
2. Klicken Sie unterhalb der Rechteverwaltung auf den Link **Template hochladen**.

   Es wird folgender Dialog geöffnet:

Abb. 3.8: Dialog Template hochladen

3. Tragen Sie in die Felder die entsprechenden Daten ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie ein bestehendes Template im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Für das Template können Sie ein Bild hinterlegen, das bei der Templateauswahl in der Rubrik angezeigt wird. Mögliche Bildformate sind die allgemein im Web verwendeten Formate `jpg/jpeg`, `png` und `gif`.

   Die hochgeladene Datei wird im Verzeichnis `\imperia\images\screenshots` mit dem Namen des Templates abgelegt.

5. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis `/site/templates` kopiert und steht anschließend in **imperia** zur Verfügung.

Hinweis:

Ähnlich wie Metadateien sollten Templates im UTF-8 Zeichensatz angelegt werden. Bei entsprechenden Vorkehrungen ist dies aber nicht zwingend notwendig. Lesen Sie hierzu auch ???

3.3.3 Template-Vorschau

Um ein Template in einer Vorschau anzeigen zu lassen, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 3.3 **Templates** auf Seite 47 beschrieben.
2. Klicken Sie in der linken Spalte der Dateiliste auf den Namen des Templates.

   Es öffnet sich ein weiteres Fenster, in dem das Template angezeigt wird. In dieser Vorschau wird kein Template-Code (IF-Abfragen, Ersetzungen, Perl-Code o.ä.) ausgeführt.

3.3.4 Template bearbeiten

Um ein Template zu bearbeiten, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 3.3 **Templates** auf Seite 47 beschrieben.
2. Betätigen Sie den Button **Bearbeiten** in der entsprechenden Zeile der Dateiliste. Damit rufen Sie den internen Editor auf, in dem Sie den Templatecode verändern können.

3.3.5 Template löschen

Um ein Template zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 3.3 **Templates** auf Seite 47 beschrieben.
2. Klicken Sie auf den Button **Löschen** neben dem entsprechenden Eintrag in der Dateiliste. Daraufhin erscheint eine Sicherheitsabfrage, die Sie mit **Okay** bestätigen müssen. Anschließend wird das Template aus dem System und damit auch aus dem Verzeichnis `/site/templates` gelöscht. Der Button **Zurück** bricht die Aktion ab.

3.4 Flexmodule

Mit dem Flexmodul-Management verwalten Sie Flexmodule. Über diesen Dialog können Sie Flexmodule bearbeiten, löschen, anlegen und aus dem lokalen Dateisystem in das **imperia** -System laden.

Das Flexmodul-Management öffnen Sie über **Menü** -> **Struktur** -> **Flexmodule**. Sie gelangen zu folgendem Dialog:

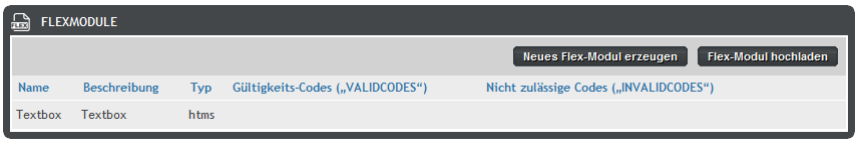


Abb. 3.9: Dialog Flexmodule bearbeiten

Der Dialog zeigt in einer Liste alle im System vorhandenen Flexmodule. Zu jedem Flexmodul werden in Spalten folgende Informationen angezeigt:

Name

Diese Spalte enthält den Dateinamen des Flexmoduls.

Typ

Abhängig davon, ob das Flexmodul in HTML- oder Perl-Code geschrieben ist, kann in dieser Spalte entweder *htms* oder *perl* stehen.

Beschreibung

Hier wird die Beschreibung angezeigt, die das Feld `DESCRIPTION` im Quellcode des Flexmoduls enthält.

Anzeigen bei ...

Diese Spalte enthält etwaige VALIDCODES des Flexmoduls. Lesen Sie hierzu auch den Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ablehnen bei ...

Hier werden etwaige INVALIDCODES des Flexmoduls angezeigt. Lesen Sie hierzu Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ferner stehen Buttons und Links zur Verfügung, mit deren Hilfe Flexmodule editiert oder gelöscht sowie neu erstellt und in das System hochgeladen werden können.

3.4.1 Neues Flexmodul erzeugen

Um ein neues Flexmodul zu erzeugen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Flexmodul-Management** wie unter Abschnitt 3.4 **Flexmodule** auf Seite 50 beschrieben.
2. Klicken Sie auf den Link **Neues Flexmodul erzeugen**. Es öffnet sich folgender Dialog:

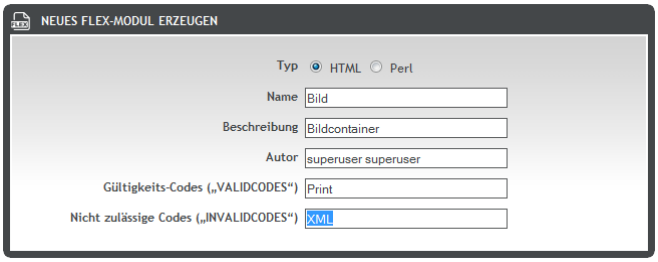


Abb. 3.10: Dialog Neues Flexmodul erzeugen

3. Tragen Sie die entsprechenden Informationen in die einzelnen Felder ein. Die Felder "Gültigkeits-Codes" und "Nicht zulässige Codes" nehmen VALID- bzw. INVALIDCODES auf, mit denen Sie die Verfügbarkeit von Flexmodulen in Templates steuern. Für weitere Einzelheiten lesen Sie den Abschnitt **Zugriff auf Flexmodule einschränken** im Kapitel **Flexmodule** des Programmierhandbuchs.
4. Klicken Sie auf den Button **Okay**, um Ihre Eingaben zu speichern. Der Button **Schließen** schließt den Dialog, ohne Eingaben zu speichern.

3.4.2 Flexmodule hochladen

Flexmodule können Sie auch außerhalb von **imperia** mit einem beliebigen Texteditor anlegen und in das System hochladen. Gehen Sie dazu wie folgt vor:

1. Rufen Sie das **Flexmodul-Management** auf wie unter Abschnitt 3.4 **Flexmodule** auf Seite 50 beschrieben.
2. Klicken Sie auf den Link **Flexmodul hochladen**. Folgender Dialog öffnet sich:

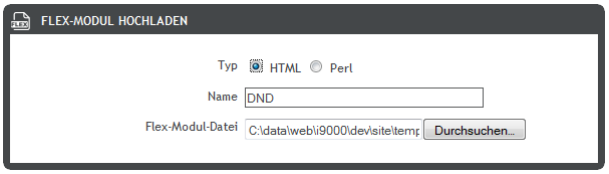


Abb. 3.11: Dialog Flexmodul hochladen

3. Tragen Sie die entsprechenden Informationen in die Felder ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie ein bestehendes Flexmodul im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis `/site/flex` kopiert und steht anschließend in **imperia** zur Verfügung.

Hinweis:

Wenn Sie ein extern erzeugtes Flexmodul im Verzeichnis `site/flex` *speichern, ohne es mit Hilfe des hier beschriebenen Dialogs hochzuladen, erscheint es ebenfalls automatisch in der Liste der Flexmodule.*

Hinweis:

Beachten Sie, dass Flexmodule im UTF-8 Zeichensatz (ohne BOM) erstellt sein müssen.

3.4.3 Flexmodule bearbeiten

Flexmodule können Sie mit jedem Text-Editor bearbeiten. Sie liegen im Verzeichnis `/site/flex` und haben je nach Inhalt die Dateiendung `.htms` (Flexmodule mit HTML-Code) oder `.perl`, bzw. `.pl` (Flexmodule mit Perl-Code).

Zum direkten Editieren in **imperia** steht der interne Editor zur Verfügung. Um ein Flexmodul mit dem internen Editor zu editieren, gehen Sie wie folgt vor:

1. Öffnen Sie das **Flexmodul-Management** wie unter Abschnitt 3.4 **Flexmodule** auf Seite 50 beschrieben.
2. Klicken Sie auf den Button **Bearbeiten** rechts neben dem entsprechenden Flexmodul.

   Es wird der interne Editor gestartet. Zur Bedienung des internen Editors lesen Sie bitte Kapitel **Der interne Editor** im Autorenhandbuch.
3. Ändern Sie das Flexmodul und speichern Sie die Änderungen.

3.4.4 Flexmodule löschen

Um ein Flexmodul zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Flexmodul-Management** wie unter Abschnitt 3.4 **Flexmodule** auf Seite 50 beschrieben.

2. Klicken Sie auf den Button **Löschen** rechts neben dem Flexmodul, das gelöscht werden soll.

   Das Flexmodul wird aus der Liste und aus dem Verzeichnis `/site/flex` entfernt.

3.5 Slotmodule

Mit dem Slotmodul-Management verwalten Sie Slotmodule. Über diesen Dialog können Sie Slotmodule bearbeiten, löschen, anlegen und aus dem lokalen Dateisystem in das **imperia** -System laden.

Hinweis:

Einzelheiten zur Syntax und zur Handbabung von Slotmodulen finden Sie im Programmierhandbuch im Kapitel **Flexmodule und Slots**.

Das Slotmodul-Management öffnen Sie über **Menü** -> **Struktur** -> **Slotmodule**. Sie gelangen zu folgendem Dialog:

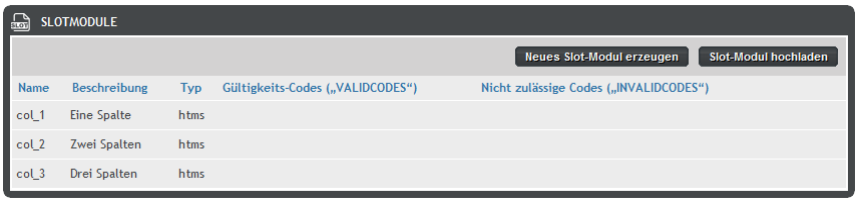


Abb. 3.12: Dialog Slotmodule bearbeiten

Der Dialog zeigt in einer Liste alle im System vorhandenen Slotmodule. Zu jedem Slotmodul werden in Spalten folgende Informationen angezeigt:

Name

Diese Spalte enthält den Dateinamen des Slotmoduls.

Typ

Abhängig davon, ob das Slotmodul in HTML- oder Perl-Code geschrieben ist, steht in dieser Spalte entweder *htms* oder *perl*.

Beschreibung

Hier sehen Sie die Beschreibung, die das Feld `DESCRIPTION` im Quellcode des Slotmoduls enthält.

Anzeigen bei ...

Diese Spalte enthält etwaige VALIDCODES des Slotmoduls. Lesen Sie hierzu auch den Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ablehnen bei ...

Hier sehen Sie etwaige INVALIDCODES des Slotmoduls. Lesen Sie hierzu auch den Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ferner stehen Buttons und Links zum Editieren, Löschen, neu Erstellen und Hochladen von Slotmodulen zur Verfügung.

3.5.1 Neues Slotmodul erzeugen

Um ein neues Slotmodul zu erzeugen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Slotmodul-Management** wie unter Abschnitt 3.5 **Slotmodule** auf Seite 52 beschrieben.
2. Klicken Sie auf den Link **Neues Slotmodul erzeugen**. Es öffnet sich folgender Dialog:

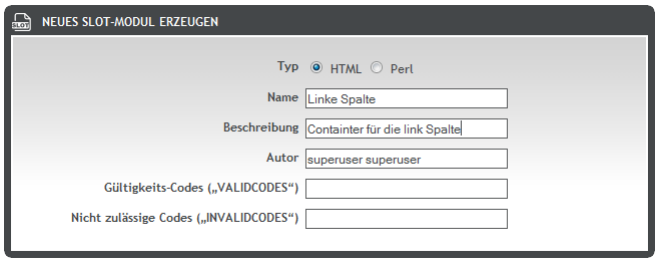


Abb. 3.13: Dialog Neues Slotmodul erzeugen

3. Tragen Sie die entsprechenden Informationen in die einzelnen Felder ein. Die Felder "Gültigkeits-Codes" und "Nicht zulässige Codes" nehmen VALID- bzw. INVALIDCODES auf, mit denen Sie die Verfügbarkeit von Slotmodulen in Templates steuern. Für weitere Einzelheiten lesen Sie den Abschnitt **Zugriff auf Slotmodule einschränken** im Kapitel **Slotmodule** des Programmierhandbuchs.
4. Klicken Sie auf den Button **Okay**, um Ihre Eingaben zu speichern. Der Button **Schließen** schließt den Dialog, ohne Eingaben zu speichern.

3.5.2 Slotmodule hochladen

Slotmodule können Sie auch außerhalb von **imperia** mit einem beliebigen Texteditor anlegen und in das System hochladen. Gehen Sie dazu wie folgt vor:

1. Rufen Sie das **Slotmodul-Management** auf wie unter Abschnitt 3.5 **Slotmodule** auf Seite 52 beschrieben.
2. Klicken Sie auf den Link **Slotmodul hochladen**. Folgender Dialog öffnet sich:

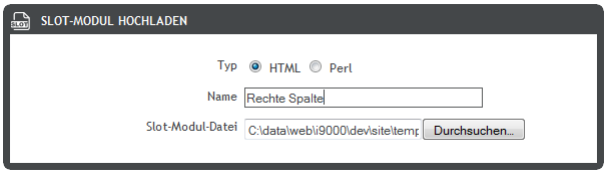


Abb. 3.14: Dialog Slotmodul hochladen

3. Tragen Sie die entsprechenden Informationen in die Felder ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie ein bestehendes Slotmodul im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis `/site/Slot` kopiert und steht anschließend in **imperia** zur Verfügung.

Hinweis:

Wenn Sie ein extern erzeugtes Slotmodul im Verzeichnis `site/Slot` *speichern, ohne es mit Hilfe des hier beschriebenen Dialogs hochzuladen, erscheint es ebenfalls automatisch in der Liste der Slotmodule.*

Hinweis:

Beachten Sie, dass Slotmodule im UTF-8 Zeichensatz (ohne BOM) erstellt sein müssen.

3.5.3 Slotmodule bearbeiten

Slotmodule können Sie mit jedem Text-Editor bearbeiten. Sie liegen im Verzeichnis `/site/Slot` und haben je nach Inhalt die Dateiendung `.htms` (Slotmodule mit HTML-Code) oder `.perl`, bzw. `.pl` (Slotmodule mit Perl-Code).

Zum direkten Editieren in **imperia** steht der interne Editor zur Verfügung. Um ein Slotmodul mit dem internen Editor zu editieren, gehen Sie wie folgt vor:

1. Öffnen Sie das **Slotmodul-Management** wie unter Abschnitt 3.5 **Slotmodule** auf Seite 52 beschrieben.

2. Klicken Sie auf den Button **Bearbeiten** rechts neben dem entsprechenden Slotmodul.

   Es wird der interne Editor gestartet. Zur Bedienung des internen Editors lesen Sie bitte Kapitel **Der interne Editor** im Autorenhandbuch.

3. Ändern Sie das Slotmodul und speichern Sie die Änderungen.

3.5.4 Slotmodule löschen

Um ein Slotmodul zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Slotmodul-Management** wie unter Abschnitt 3.5 **Slotmodule** auf Seite 52 beschrieben.
2. Klicken Sie auf den Button **Löschen** rechts neben dem Slotmodul, das gelöscht werden soll.

   Das Slotmodul wird aus der Liste und aus dem Verzeichnis `/site/Slot` entfernt.

3.6 Codeinclude-Dateien

In separaten Dateien gespeicherte Code-Bausteine lassen sich über die **imperia** -Benutzeroberfläche anlegen, editieren und verwalten. Sie können außerdem Code-Bausteine hochladen, die Sie außerhalb von **imperia** erstellt haben.

Mehr über die Erstellung von Codeincludes lesen Sie im Abschnitt "Codeincludes" im Programmierhandbuch.

Das Codeinclude-Management erreichen Sie, wenn Sie den Menüpunkt *Menü -> Struktur -> Codeinclude-Dateien* auswählen.

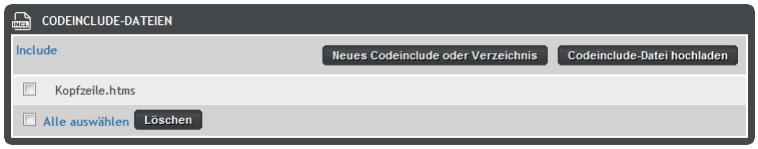


Abb. 3.15: Der Dialog Codeincludes bearbeiten

Sie erhalten eine Übersicht des Inhalts des Verzeichnisses `site/include` auf Ihrem Produktionssystem. Dabei werden die Dateinamen vollständig inklusive der Dateiendung angezeigt.

Zur besseren Übersichtlichkeit können Sie Ihre Includes in Unterverzeichnissen strukturieren. Verzeichnisnamen sind in der Liste fett dargestellt und verlinkt. Klicken Sie einen Verzeichnisnamen an, um dessen Inhalt als Dateiliste anzeigen zu lassen.

Wenn Sie sich in einem Unterverzeichnis befinden, ist der erste Eintrag in der Dateiliste immer ein Link auf die nächsthöhere Verzeichnisebene. Dieser Link ist durch zwei Punkte kenntlich gemacht.

Alle Funktionen, die Sie im Dialog "Codeincludes bearbeiten" auswählen, werden in dem Verzeichnis ausgeführt, dessen Inhalt gerade in der Übersicht angezeigt wird.

3.6.1 Neues Codeinclude / Verzeichnis erstellen

Um ein neues Codeinclude oder Verzeichnis anzulegen, führen Sie folgende Schritte aus:

1. Öffnen Sie das Codeinclude-Management wie unter Abschnitt 3.6 **Codeinclude-Dateien** auf Seite 54 beschrieben.
2. Klicken Sie auf den Link **Neues Codeinclude oder Verzeichnis**. Sie gelangen zu folgendem Dialog:

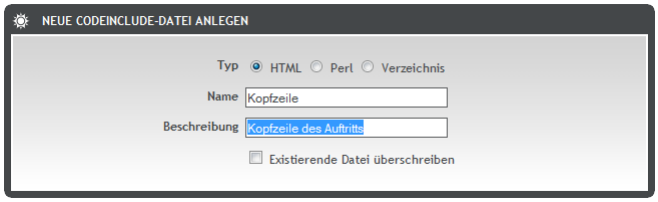


Abb. 3.16: Der Dialog Neue Codeinclude-Datei anlegen

3. Wählen Sie über die Radiobuttons aus, ob Sie ein *HTML*- oder *Perl*-Codeinclude bzw. ein *Verzeichnis* erstellen möchten.

Hinweis:

imperia *ergänzt automatisch die jeweiligen Dateiendungen für HTML- und Perl-Includes. Sie können diese also bei der Eingabe des Dateinamens weglassen.*

4. Geben Sie auch einen kurzen *Beschreibungstext* und einen *Namen* ein.
5. Bestätigen Sie Ihre Eingaben durch einen Klick auf **Okay** oder verwerfen Sie sie durch einen Klick auf **Schließen**. Wenn Sie Ihre Eingaben verwerfen, werden Sie automatisch zum Dialog **Codeincludes bearbeiten** zurückgeführt.

Sollte bereits ein Verzeichnis oder Include mit dem von Ihnen eingegebenen Namen bestehen, erscheint eine Warnmeldung, wenn Sie Ihre Eingaben mit *Okay* bestätigen.

Geben Sie dann einen anderen Namen ein oder aktivieren Sie die Option zum Überschreiben bestehender Dateien, indem Sie die Checkbox zwischen den beiden Eingabefeldern markieren.

Hinweis:

Verzeichnisse können auf diese Weise nicht überschrieben werden, um den Verlust von Include-Dateien durch das Überschreiben von Verzeichnissen, die Includes enthalten, zu verhindern.

Nachdem Sie Ihre Eingaben bestätigt haben, werden Sie automatisch zum Dialog *Codeincludes bearbeiten* zurückgeführt. Ihre neu erstellte Datei bzw. Ihr neu erstelltes Verzeichnis erscheint nun in der Übersicht.

Hinweis:

Eine mit **imperia** *neu erstellte Codeinclude-Datei ist lediglich ein Grundgerüst ohne Inhalt. Sie müssen diese Dateien noch editieren (siehe Abschnitt 3.6.3* **Codeinclude-Dateien bearbeiten** *auf Seite 57).*

3.6.2 Codeinclude-Dateien hochladen

Grundsätzlich können Sie Codeincludes mit einem beliebigen Texteditor erstellen. Außerhalb von **imperia** angelegte Codeinclude-Dateien müssen Sie in **imperia** hochladen, um Sie im System verfügbar zu machen. Führen Sie dazu folgende Schritte aus:

1. Öffnen Sie das Codeinclude-Management wie unter Abschnitt 3.6 **Codeinclude-Dateien** auf Seite 54 beschrieben.
2. Klicken Sie auf **Codeinclude-Datei hochladen**. Dieser Link führt Sie zu folgendem Dialog:

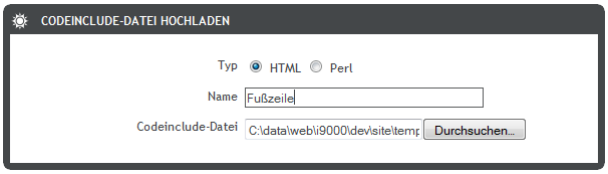


Abb. 3.17: Der Dialog Codeinclude-Datei hochladen

3. Bestimmen Sie den Typ der hochzuladenden Datei durch Klick auf einen der Radiobuttons. Zur Verfügung stehen die Typen *HTML* und *Perl*. Entsprechend Ihrer Auswahl hängt **imperia** automatisch die passende Dateiendung an den Dateinamen an.
4. Tragen Sie in das Eingabefeld *Name* einen Dateinamen für Ihre Include-Datei ein. Ohne Angabe eines Dateinamens wird der Upload-Vorgang mit einer Warnmeldung abgebrochen.

   Mit dem Button *Durchsuchen* öffnen Sie ein Dialogfenster zur Auswahl der hochzuladenden Codeinclude-Datei.

5. Bestätigen Sie Ihre Eingaben durch Klick auf den Button **Okay** oder verwerfen Sie sie, indem Sie auf **Schließen** klicken. Sie werden dann automatisch zum Dialog **Codeincludes bearbeiten** zurückgeführt.

Sollte bereits eine Codeinclude-Datei mit dem von Ihnen eingegebenen Namen bestehen, wird eine Warnmeldung angezeigt und der Upload nicht ausgeführt. Wie bei der Erstellung von Includes oder Verzeichnissen, können Sie auch hier eine Option zum Überschreiben bestehender Dateien aktivieren.

Nach dem Hochladen der Datei erscheint wieder der Dialog **Codeincludes bearbeiten** mit aktualisierter Dateiliste.

3.6.3 Codeinclude-Dateien bearbeiten

Codeincludes lassen sich folgendermaßen über die **imperia** -Benutzeroberfläche bearbeiten:

1. Rufen Sie zunächst das Codeinclude-Management auf (siehe Abschnitt 3.6 **Codeinclude-Dateien** auf Seite 54).
2. Klicken Sie auf den Bearbeiten-Button neben dem Dateinamen des Includes, das Sie bearbeiten wollen. Daraufhin öffnet sich ein Popup-Fenster mit dem internen **imperia** -Editor, in dem Sie die ausgewählte Datei editieren können. Für eine genaue Beschreibung des internen Editors lesen Sie bitte den Abschnitt **Der interne Editor** im Benutzerhandbuch.

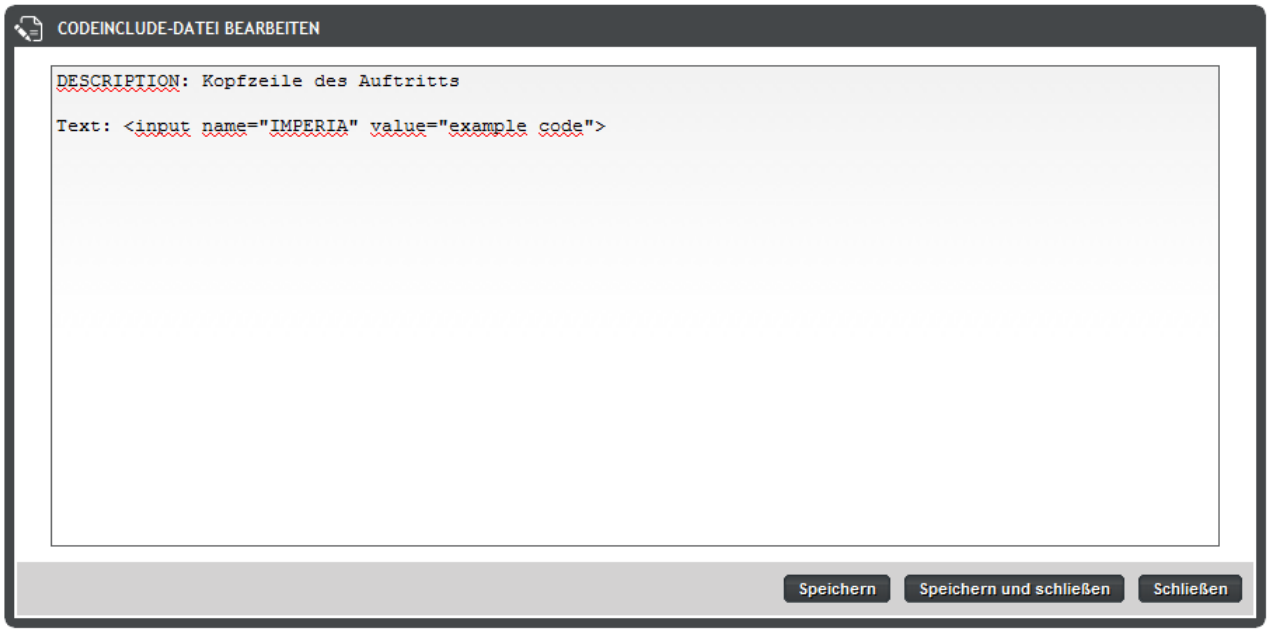


Abb. 3.18: Das Bearbeitungsfenster für Codeincludes

Sie können sowohl extern als auch mit **imperia** erstellte (siehe Abschnitt 3.6.1 **Neues Codeinclude / Verzeichnis erstellen** auf Seite 54) Codeincludes bearbeiten.

3.6.4 Codeinclude-Dateien / Verzeichnisse löschen

Führen Sie die folgenden Schritte aus, um ein Codeinclude oder ein Verzeichnis für Codeinclude-Dateien über die **imperia** -Benutzeroberfläche zu löschen:

1. Öffnen Sie das Codeinclude-Management wie unter Abschnitt 3.6 **Codeinclude-Dateien** auf Seite 54 beschrieben.
2. Klicken Sie auf den Löschen-Button neben einem Codeinclude oder Verzeichnis, um dieses zu löschen.
3. Wenn Sie mehrere Includes bzw. Verzeichnisse entfernen wollen, markieren Sie zunächst die Checkboxen links daneben und klicken Sie anschließend auf den Button Auswahl löschen „Löschen“

3.7 Dynamische Module

Dynamische Module, die in der Datei `dynamic.conf` konfiguriert werden, werden ausführlich in dem Kapitel **Dynamische Module** des Programmierhandbuchs behandelt. Um die Datei `dynamic.conf` zu editieren, steht der interne Editor zur Verfügung. Die Datei `dynamic.conf` liegt im Verzeichnis `/site/config`. Gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Dynamische Module** im Ordner **Struktur**.

   Es wird der interne Editor geöffnet und die vorhandene **dynamic.conf** wird angezeigt.

2. Ändern Sie ggfs. den Code im Textfeld.
3. Speichern Sie die Änderungen über den Button **Okay** unterhalb des Dialogs.

3.7.1 Laden einer externen Datei in den Editor

Um eine außerhalb von **imperia** erzeugte Datei in den Editor zu laden, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Dynamische Module** im Ordner **Struktur**.
2. Geben Sie den Pfad und Dateinamen der gewünschten Datei in das Feld **Datei hochladen** ein oder benutzen Sie den Button **Durchsuchen...**, um einen Dialog zum Durchsuchen Ihrer Festplatten zu öffnen.
3. Speichern Sie die Datei über den Button **Okay** am unteren Rand der **imperia** -Oberfläche.

   Die externe Datei wird jetzt als `dynamic.conf` gespeichert und die bereits vorhandene Datei wird überschrieben.

3.8 Template-Reparser

Der Template-Reparser bietet die Möglichkeit, geänderte Templates erneut zu parsen, so dass alle Dokumente, die mit einer früheren Version dieses Templates erzeugt wurden, erneut erzeugt werden.

Hierbei können sowohl Dokumente, die sich bereits auf einem Zielsystem befinden, als auch solche, die in der Freischaltliste angezeigt werden, neu erstellt werden.

Zusätzlich wird bestimmt, ob die neu erstellten Dokumente nach dem Reparsen in einen Workflow einfließen, der Freischaltliste hinzugefügt oder automatisch auf ein Zielsystem freigeschaltet werden sollen.

Um den Template-Reparser zu verwenden, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Template-Reparser** im Ordner **Struktur**.

   Es wird folgender Dialog in einem neuen Fenster geöffnet:

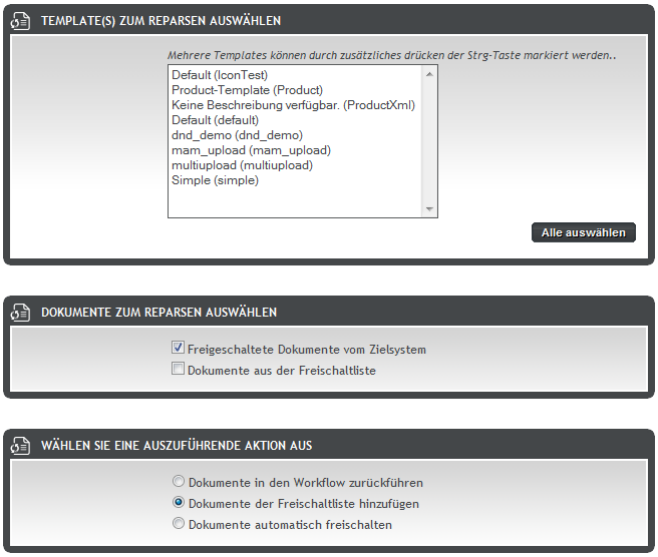


Abb. 3.19: Dialog Template-Reparser

Der Dialog des Template-Reparsers gliedert sich in drei Abschnitte:

Template zum Reparsen auswählen

Hier werden die Templates ausgewählt, die neu geparst werden sollen.

Dokumente zum Reparsen auswählen

In diesem Abschnitt wird ausgewählt, ob die Dokumente vom Zielsystem und / oder die aus der Freischaltliste neu erstellt werden sollen.

Aktion nach dem Reparsen

In diesem Abschnitt wird bestimmt, wie mit den neu erstellten Dokumenten verfahren werden soll. Es kann zwischen den Aktionen **Dokumente in den Workflow zurückführen**, **Dokumente der Freischaltliste hinzufügen** und **Dokumente automatisch freischalten** gewählt werden.

2. Markieren Sie im Abschnitt **Template zum Reparsen auswählen** die Templates, die neu geparst werden sollen. Mehrere Templates werden durch zusätzliches Drücken der Strg-Taste markiert.

   Der Link **Alle auswählen** markiert alle Templates in der Liste.

Achtung:

Wenn Sie alle Templates markieren, dauert das Reparsen, wenn viele Dokumente neu erstellt werden müssen, unter Umständen sehr lange. Infolgedessen kann es zu Server-Timeouts und somit zum vorzeitigen Abbruch des Vorganges kommen.

Für das Reparsen größerer Dokumentenmengen steht ein Kommandozeilen-Tool zur Verfügung, das in Abschnitt 2.14 **Reparsen umfangreicher Sites** *auf Seite 37 beschrieben ist.*

3. Wählen Sie im Abschnitt **Dokumente zum Reparsen auswählen**, welche Dokumente neu erstellt werden sollen.
4. Wählen Sie im Abschnitt **Aktion nach dem Reparsen** die Aktion aus, die nach dem Reparsen ausgeführt werden soll.
5. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

   Das Reparsen wird gestartet und die Dokumente entsprechend Ihrer Auswahl weiterverarbeitet.

3.9 Assets verschieben

Diese Funktion ist ausführlich in ??? beschrieben.

3.10 Systemdateien abgleichen

Mit dieser Funktion können Sie geänderte Systemdateien auf verschiedenen Zielsysteme übertragen. Dies ist dann notwendig, wenn beispielsweise Systemdienste, die den Hintergrund-Daemon auf einem Zielsystem betreffen (SiteActive), geändert wurden.

Gehen Sie wie folgt vor:

1. Legen Sie eine Datei `/site/config/system_files.conf` an und tragen Sie dort die zu kopierenden Dateien inklusive des Pfades ausgehend vom SITE-Verzeichnis ein. Im Standard wird dort die Datei **/config/imperia.schedule** angegeben, in der die Systemdienste abgespeichert werden.
2. Klicken Sie im Ordner **Struktur** auf den Menüpunkt **Systemdateien abgleichen**.

   Es öffnet sich der Dialog **Systemdateien kopieren** in einem neuen Fenster.
3. Markieren Sie in der Liste **Vorhandene Zielsysteme** die Zielsysteme, auf die die geänderten Systemdateien übertragen werden sollen.
4. Klicken Sie auf den Button **Okay**. Der Abgleich wird durchgeführt und es wird ein Protokoll des Vorgangs angezeigt. Der Button **Schließen** schließt den Dialog, ohne dass ein Abgleich durchgeführt wird.

3.11 Dateirechte in imperia verwalten

imperia bietet die Möglichkeit, einzeln Rechte pro Template-, Meta- und Workflow-Datei zu vergeben. Hierzu werden einer Rolle Schreib-, Lösch- oder Leserechte für jede Datei einzeln zugewiesen oder entzogen. Diese Funktion kann gleichzeitig für mehrere Rollen und mehrere Dateien eines Typs durchgeführt werden. Im Folgenden beschreiben wir die nötigen Schritte am Beispiel der Metadateien. Für alle anderen Dateitypen sind die Schritte analog durchzuführen.

Um die Rechte für **imperia** -Dateien zu vergeben, gehen Sie wie folgt vor:

1. Öffnen Sie den Management-Dialog für die zu editierenden Dateien.
2. Klicken Sie auf das Pluszeichen mit der Beschriftung **Rechte** unterhalb der Dateiliste.

   Folgender Dialog öffnet sich:

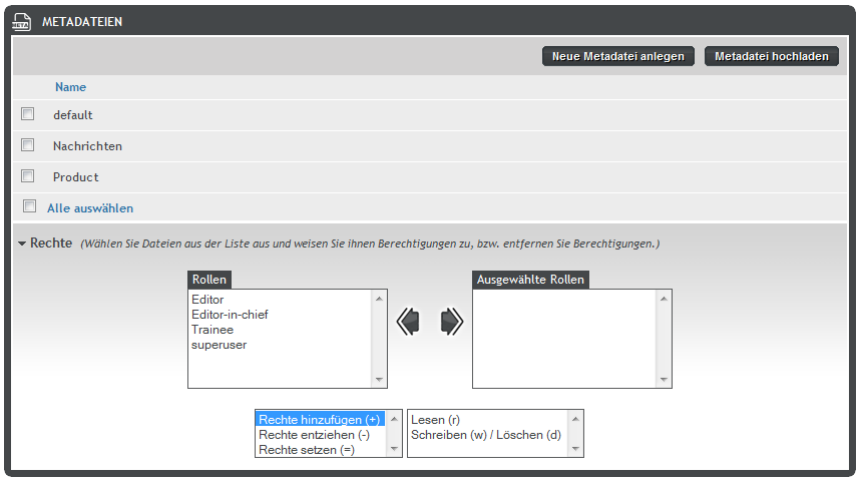


Abb. 3.20: Dialog Rechtevergabe, Beispiel Metadateien

Der Dialog gliedert sich in folgende Abschnitte:

Dateiliste

Die Dateiliste wird in dieser Ansicht um Checkboxen ergänzt. Einzelne Dateien werden mit Hilfe der jeweiligen Checkbox markiert, alle Dateien können mit einem Klick über die Checkbox "Alle auswählen" markiert werden.

Rechte

In diesem Abschnitt werden die Rechte für die Rollen vergeben. Der Abschnitt ist in drei Felder unterteilt. Das linke Feld enthält die Liste aller im System eingerichteten Rollen, auf die man als angemeldeter Benutzer Zugriff hat. Das mittlere Feld enthält alle verfügbaren Funktionen und das rechte Feld alle Rechte, die vergeben werden können.

3. Markieren Sie in der Dateiliste mit Hilfe der Checkboxen die Dateien, für die Rechte gesetzt werden sollen.
4. Markieren Sie im Abschnitt **Rechte** in der linken Liste die Rollen, deren Rechte geändert bzw. gesetzt werden sollen. Mehrere Einträge werden mit Hilfe der Strg-Taste markiert.
5. Wählen Sie im Abschnitt **Rechte** in der mittleren Liste die Funktion aus, die Sie ausführen wollen. Die Funktionen bedeuten im Einzelnen:

Rechte hinzufügen (+)

Diese Funktion fügt die ausgewählten Rechte den bereits bestehenden Rechten der ausgewählten Rollen hinzu.

Rechte entziehen (-)

Hiermit werden die markierten Rechte den markierten Rollen entzogen.

Rechte setzen (=)

Diese Funktion setzt die Rechte der ausgewählten Rollen auf die markierten Rechte. Nicht markierte Rechte werden den Rollen entzogen.

6. Markieren Sie in der rechten Liste die Rechte, die geändert bzw. eingestellt werden sollen.
7. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

3.12 imperia Onlinehilfe

Die Dokumentation für die Onlinehilfe liegt im Verzeichnis

```
/imperia/documentation/de/imperia.
```

Zusatzmodule werden ihre Dokumentation unter

```
/imperia/documentation/de/MODULNAME
```

ablegen.

Derzeit gibt es nur eine deutsche **imperia** Dokumentation. Weitere Sprachen können aber von der Onlinehilfe verarbeitet werden. Zuerst wird versucht das entsprechende Dokument in der Benutzersprache zu laden. z.B. 'bg'. (`/imperia/documentation/bg/imperia/`). Ist dieses Dokument nicht vorhanden, so wird versucht die englische Variante zu laden (`/imperia/documentation/en/imperia/`. Schlägt auch dies fehl, so wird die deutsche Dokumentation (`/imperia/documentation/de/imperia/` angezeigt.

Kapitel 4. iWE - imperia -WYSIWYG-Editor

Der iWE (**imperia** -WYSIWYG-Editor) ist ein WYSIWYG Editor auf Basis des FCK-Editors der durch entsprechende Anpassungen auf die Anforderungen des **imperia** CMS angepasst wurde.

Aktuelle Informationen zum FCK finden Sie unter *www.fckeditor.net* (Stand der Quellenangabe: Juli 2009).

4.1 Migration vom EWE-Modul zum iWE

Wenn Sie bisher den Easy Web Editor (EWE) als WYSIWYG-Editor verwendet haben, können Sie den damit erzeugten Inhalt durch den iWE bearbeiten. Eine Änderung im Template-Code ist hierfür nicht notwendig. Der iWE verfügt über einen Kompatibilitätsmodus, der die EWE-Template-Syntax einschließlich vieler Aufrufparameter unterstützt. Die Kompatiblität kann je nach Implementierung variieren. Fügen Sie zur Aktivierung des EWE-Kompatibilitätsmodus folgende Variable in der Datei `site/config/system.conf` hinzu:

```
"EWE_COMPAT_MODE" = "1"
```

Durch den „EWE_COMPAT_MODE“ werden EWE-Templateaufrufe als iWE-Aufrufe interpretiert.

Außerdem müssen Sie das EWE-Modul deaktivieren. Kommentieren Sie dazu in der Datei `site/config/system.conf` folgenden Eintrag aus:

```
"TEMPLATE_EWE" = "yes"
```

Der **imperia** Template-Prozessor stellt nun anstelle des EWE den iWE dar.

4.2 Konfiguration des iWE

Der iWE bietet eine Vielzahl von Konfigurationsoptionen, mit welchen die Funktionalitäten, das Aussehen und das Verhalten beeinflusst werden.

Es gibt es zwei Möglichkeiten den Editor zu konfigurieren:

- Konfiguration im Template

  Bei dieser Methode erfolgt die Konfiguration der jeweiligen Instanz des Editors über das **imperia** -Template.

  Die Verwendung des iWE im Template können Sie im, Kapitel **iWE - imperia -WYSIWYG-Editor** im Programmierhandbuch nachschlagen.

- Steuerung über eine Konfigurationsdatei

  Hierbei wird entweder für jede Instanz eine eigene Konfigurationsdatei im **imperia** -Template spezifiziert, oder es wird eine globale Datei für alle Instanzen verwendet.

Wichtig:

Bitte beachten Sie die Hierarchie der Konfigurationsmöglichkeiten. Wenn Sie eine Konfigurationsoption im Template gesetzt haben, hat diese grundsätzlich Vorrang vor der gesetzten Option in der Konfigurationsdatei. Es gibt also insgesamt drei Hierachiestufen:

1. *Template*
2. *Benutzerdefinierte Konfigurationsdatei der Instanz*
3. *Globale Konfigurationsdatei*

Die globale Konfigurationsdatei finden im Dateisystem unter:

```
DOCUMENT-ROOT/imperia/fckeditor/fckconfig.js
```

4.3 Konfigurationsoptionen

Der iWE bietet weitere Konfigurationsoptionen. Es wird unterschieden zwischen:

- Template-Only Optionen

  Dabei handelt es sich um Konfigurationsoptionen, welche nur im Template gesetzt werden können. Diese Optionen sind speziell für die Verwendung des Editors in **imperia** .

- Freie Optionen

  Dies sind Optionen (FCK-Optionen), welche sowohl im Template als auch in den Konfigurationsdateien verwendet werden können.

4.3.1 Template-Only Optionen

Template-Only Optionen haben die Besonderheit, dass sie, wie der Name bereis vermuten lässt, nur in der Processing Instruction im Template gesetzt werden können. Folgende Optionen gibt es:

- Width

  Mit der Option „Width“ definieren Sie die Breite des Editorfensters.

  Mögliche Werte sind numerische Werte in Pixel. Beispiel:

```
<?imperia iwe
    id: iwe1
    Width: 570
?>
```

- Height

  Mit der Option „Height“ definieren Sie die Höhe des Editorfensters.

  Mögliche Werte sind numerische Werte in Pixel. Beispiel:

```
<?imperia iwe
    id: iwe1
    Height: 400
?>
```

- External

  Die Option „External“ schaltet den External-Mode an bzw. ab. Ein aktivierter External-Mode führt dazu, dass der Editor in einem eigenen Popup-Fenster geöffnet wird und nicht eingebettet im Template erscheint.

  Mögliche Werte sind: '1' (an) oder '0' (aus). Beispiel:

```
<?imperia iwe
    id: iwe1
    External: 1
?>
```

- ExternalStyle

  Die Option „ExternalStyle“ definiert das Aussehen des Platzhalters im Template, welcher den Inhalt anzeigt, der über den Editor im External-Mode gepflegt wird. Als möglicher Wert kann CSS-Code verwendet werden. Ebenso ist die Angabe einer CSS-Datei möglich, welche die Klasse 'imperia_external_editor' beinhaltet. Beispiel:

```
<?imperia iwe
    id: iwe1
    External: 1
    ExternalStyle: background: red
?>
```

- Content

  Die Option 'Content' bietet die Möglichkeit, die jeweilige Instanz des iWE vorab mit Inhalt zu befüllen.

  Ein möglicher Wert ist jede Art von Text (String) oder eine **imperia** -Variable. Beispiel:

```
<?imperia iwe
    id: iwe1
    Content: Das ist ein Beispiel
?>
```

4.3.2 Freie Optionen

Der iWE bietet eine große Menge an Möglichkeiten, die Funktionalität an eigene Bedürfnisse anzupassen. Im folgenden Abschnitt werden einige der wichtigsten Optionen genannt, jedoch nicht alle. Eine komplette Übersicht über alle verfügbaren Konfigurationsoptionen finden Sie im Internet unter:

http://docs.cksource.com/FCKeditor_2.x/Developers_Guide/Configuration/Configuration_Options

Im folgenden Abschnitt wird nur die Syntax für die Verwendung in einer Konfigurationsdatei verwendet.

4.3.2.1 Verhalten des Editors

- BaseHref

  Die Option 'BaseHref' funktioniert wie der HTML-Tag '<base href="">'. Sie definiert die Basis-URL aller verwendeten Links und Bilder.

  Als möglicher Wert kann eine URL gesetzt werden. Beispiel:

```
FCKConfig.BaseHref = 'http://www.mysite.com/';
```

- DocType

  Die Option 'DocType' ermöglicht die Angabe des DOCTYPE, welches in dem editierbaren Teil verwendet wird. Er spezifiziert wie gerendert wird.

  Als gültigen Wert ist eine DOCTYPE-Syntax anzugeben. Das folgende Beispiel lässt den Editor den eingegebenen Inhalt als 'XHTML 1.0 Transitional' rendern:

```
FCKConfig.DocType = '<!DOCTYPE html PUBLIC
"-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">';
```

- MaxUndoLevels

Die Option 'MaxUndoLevels' setzt die maximale Anzahl von 'Undo' bzw. 'Rückgängig' Schritten, die der Benutzer nutzen kann. Der Standardwert ist 15.

Beispiel:

```
FCKConfig.MaxUndoLevels = 20;
```

- StartupFocus

  Wenn die Option 'StartupFocus' auf 'true' gesetzt wird, erzwingt der Editor, dass der Eingabecursor innerhalb des Eingabebereichs erscheint. Dies ist sinnvoll, wenn sich nur eine Instanz des iWE im Template befindet und der Benutzer direkt mit dem Tippen beginnt.

  Als mögliche Werte stehen 'true' oder 'false' zur Auswahl. Beispiel:

```
FCKConfig.StartupFocus = true;
```

4.3.2.2 Styles

- BodyId und BodyClass

  Dieses Optionspaar ermöglicht die Angabe einer ID und/oder einer Klasse, welche zum BODY des zu editierenden Bereichs hinzugefügt wird.

  Beispiel: Ihre Hintergrundfarbe (background-color) ist schwarz (#000), doch der Inhalt, welcher über den Editor gepflegt wird, befindet sich in einem ' ' Element und nutzt dieses CSS:

```
body { background-color:#000; } h1 { color:#FFF; }
#news { background-color:#FFF; } #news h1 { color:#333; }
```

  Wenn Sie also in Ihrer Konfigurationsdatei die folgende Zeile verwenden:

```
FCKConfig.BodyId = 'news' ;
```

  wird der Bereich, welcher durch den Editor gepflegt wird, einen weissen Hintergrund haben. Ebenfalls werden alle H1-Elemente in dunkler Farbe dargestellt. Diese Option bietet also die Möglichkeit, die gleiche CSS-Datei für die gesamte Seite und für den Editor zu verwenden.

- CustomStyles

  Die Option 'CustomStyles' bietet Ihnen die Möglichkeit, eigene Styles in das 'Styles'-Menue im Editor hinzuzufügen. Das folgende Beispiel zeigt, wie ein neuer Style angelegt wird.

  Beispiel: Sie wollen einen neuen Style hinzufügen, welcher 'MY STYLE 1' heisst und folgende Eigenschaften hat:

  - Der Style beinhaltet das Format Überschrift 2 (h2)
  - Die Schriftfarbe ist blau (Blue)
  - Die Hintergrundfarbe ist rot (Red)

  Um den benutzerdefinierten Style 'MY STYLE 1' hinzuzufügen, benötigen Sie den folgenden Code in Ihrer Konfigurationsdatei:

```
FCKConfig.CustomStyles = { 'MY STYLE 1' : {Element
:'h2', Styles : {'color' : 'Blue' , 'background-color' : 'Red' } }};
```

  Wie Sie sehen, ist die Syntax der Option recht einfach:

  - Innerhalb eines Apostrophenpaares (' ') wird der Name des Styles gesetzt.

- Nach dem Doppelpunkt wird die Definition des Styles in geschweifte Klammern ({ }) geschrieben.
- 'Element' zeigt auf das Element, für welches der Style übernommen werden werden soll (z.B. h1, h2, h3).
- 'Styles' definiert, welche Eigenschaften Ihr benutzerdefinierter Style haben soll (color, background-color, font-family).

- FontColors

  Die Option 'FontColors' definiert die Farben, die im 'Text Color' und im 'Background Color' Menue aufgelistet werden.

  Als gültige Werte können HTML-Farbcodes im HEX Format angegeben werden (FFFFFF). Beispiel:

```
FCKConfig.FontColors =
'000000,993300,333300,003300,003366,000080,333399,333333,800000,
FF6600,808000,808080,008080,0000FF,666699,808080,FF0000,FF9900,
99CC00,339966,33CCCC,3366FF,800080,999999,FF00FF,FFCC00,FFFF00,
00FF00,00FFFF,00CCFF,993366,C0C0C0,FF99CC,FFCC99,FFFF99,CCFFCC,
CCFFFF,99CCFF,CC99FF,FFFFFF';
```

- FontFormat

  Die Option 'FontFormat' definiert die Schriftformate (z.B. Überschrift 1, Überschrift 2), welche im 'Format' Menue aufgelistet sind.

  Als gültige Werte können HTML-Formate angegeben werden (z.B. p, h1, div). Beispiel:

```
FCKConfig.FontFormats ='p;h1;h2;h3;h4;h5;h6;pre;address;div';
```

- FontNames

  Die Option 'FontFormat' definiert die Schriftarten, welche im 'Font' Menue aufgelistet sind.

  Beispiel:

```
FCKConfig.FontNames= 'Arial;Comic Sans MS;Courier New;
Tahoma;Times New Roman;Verdana';
```

- FontSizes

  Die Option 'FontSizes' definiert die Schriftgrößen, welche im 'Size' Menue aufgelistet sind. Standardmässig sind folgende Werte verfügbar:

```
FCKConfig.FontSizes =
'smaller;larger;xx-small;x-small;small;medium;large;x-large;xx-large';
```

  Es kann jeder gültige CSS-kompatible Wert genutzt werden. Das folgende Beispiel setzt Pixel, 'em' und Prozent-Größen:

```
FCKConfig.FontSizes = '14px;1.5em;80%';
```

  Für jede Größe kann auch ein Name vergeben werden, der anstatt des Wertes im 'Size' Menue angezeigt wird. Name und Wert müssen durch einen Schrägstrich getrennt werden. Beispiel:

```
FCKConfig.FontSizes = '14px/Fourteen
            Pixels;1.5em/A bit biggger;80%/A bit smaller' ;
```

4.3.2.3 HTML Ausgabe

- FillEmptyBlocks

  Dieser Parameter füllt leere Tabellenzellen mit "Whitespaces" auf, um einen Zellenumbruch zu verhindern. Er ist standardmäßig gesetzt.

```
FCKConfig.FillEmptyBlocks = false
            ;
```

4.3.2.4 Benutzer-Interface

- AutoDetectLanguage

  Die Option 'AutoDetectLanguage' gibt dem Editor die Möglichkeit, die Systemsprache zu erkennen. Wenn die Option auf 'true' gesetzt ist, wird der Editor in der selben Sprache, wie die des Browsers oder des Betriebsystems, angezeigt.

  Als gültiger Wert kann 'true' oder 'false' gesetzt werden.

```
FCKConfig.AutoDetectLanguage = true;
```

- DefaultLanguage

  Die Option 'DefaultLanguage' spezifiziert die Sprache, welche vom iWE als Standard verwendet wird.

  Gültige Werte sind die jeweiligen Landeskennungen (z.B.: Deutsch (de), Franzoesisch (fr), Italienisch (it), usw.)

```
FCKConfig.DefaultLanguage = "de";
```

- Pluging.add

  Mit der Option 'Plugin.add' wird ein neuer Eintrag in das Kontextmenü des Editors eingefügt.

  Im folgendem Beispiel werden Einträge zum Einfügen von Tabellen in das Kontextmenü eingebettet.

```
FCKConfig.Plugins.Add( 'tbrow', null, sPluginPath );
```

- ToolbarSets

  Mit dieser Option ist es möglich, die Symbolleiste (Toolbar) entsprechend anzupassen.

  Nach einer initialen Installation des iWE sieht die Toolbar wie folgt aus:

```
FCKConfig.ToolbarSets["Imperia"] = [
['Source','-','Preview','-','Templates'],
['Cut','Copy','Paste','PasteText','PasteWord','-','Print','SpellCheck'],
['Undo','Redo','-','Find','Replace','-','SelectAll','RemoveFormat'],
['Form','Checkbox','Radio','TextField','Textarea','Select','Button',
    'MAMAssetButton','HiddenField'],
'/',
['Bold','Italic','Underline','StrikeThrough','-','Subscript','Superscript'],
['OrderedList','UnorderedList','-','Outdent','Indent','Blockquote','CreateDiv'],
['JustifyLeft','JustifyCenter','JustifyRight','JustifyFull'],
['MAMAssetLink','Unlink','Anchor'],
['MAMAsset','Table','Rule','Smiley','SpecialChar','PageBreak'],
'/', ['Style','FontFormat','FontName','FontSize'],
['TextColor','BGColor'], ['FitWindow','ShowBlocks']];
```

Mit dieser Option besteht zum Beispiel die Möglichkeit, Einträge zum Einfügen von Tabellen hinzu zufügen, wie im folgendem Beispiel aufgeführt.

```
FCKConfig.ToolbarSets['TableTest'] = [
['tbrow','tbcol','tbtab']
] ;
```

Kapitel 5. Rubriken

Mit Hilfe von Rubriken wird ein Projekt in verschiedene Themenbereiche unterteilt. Die am häufigsten verwendete Unterteilungsstrategie ist die Differenzierung nach Themen- oder Sachgebieten. Jedem Themenbereich wird eine eigene Rubrik zugeordnet, in der wiederum Unterrubriken für einzelne Themen erstellt werden. Unterrubriken können beliebig verschachtelt werden.

Beispielsweise kann eine Rubrik *Sport*, eine Rubrik *Reisen* und eine Rubrik *News* erstellt werden. Als Unterrubriken für die Hauptrubrik *Sport* könnten zum Beispiel die Rubriken *Segeln*, *Golf* und *Fußball* existieren. Die Unterrubrik *Fußball* könnte weiter unterteilt werden in *Bundesliga* und *Champions League*.

Rubriken können entweder nur weitere Unterrubriken oder Unterrubriken und Dokumente enthalten. Rubriken, die nur Unterrubriken enthalten können, werden zur Strukturierung des Rubrikenbaums verwendet. Wird diese Eigenschaft von **enthält Unterrubriken und Dokumente** zu **enthält Unterrubriken** geändert, kann dies zu folgendem Problem führen:

- Es können keine Dokumente mehr aus dem Archiv oder vom Zielsystem in diese Rubrik importiert werden. Der importierende Benutzer erhält eine Fehlermeldung (Keine Berechtigung, um Dokumente in dieser Rubrik zu erzeugen!).

Da im Allgemeinen nicht jeder Benutzer Dokumente in jeder Rubrik erstellen oder bearbeiten soll, werden beim Erzeugen einer Rubrik die Zugriffsrechte bestimmt. Das heißt, es wird festgelegt, welche Rolle einem Benutzer Zugriffsrechte auf die Rubrik verleiht.

Wählt ein Benutzer den Menüpunkt **Neues Dokument erstellen** aus, sieht er im folgenden Dialog nur die Rubriken, in denen er aufgrund seiner aktuellen Rolle Dokumente erstellen darf. Wichtig hierbei ist, dass er auf alle Oberrubriken dieser Rubriken Leserechte haben muss.

Zugriffsrechte können in vier Abstufungen erteilt werden:

- Lesen (Inhalt der Rubrik anzeigen lassen),
- Dokumente erzeugen,
- Dokumente löschen und
- Rubrik bearbeiten.

Weiterhin werden folgende Eigenschaften eines Dokuments durch die Rubrik bestimmt:

- die standardmäßig verwendete Metadatei,
- das standardmäßig verwendete Template und
- der verwendete Workflow.

5.1 Rubriken-Eigenschaften bestimmen

Die Eigenschaften einer Rubrik stellen Sie im Dialog **Rubriken-Eigenschaften** ein. Um diesen Dialog zu öffnen wählen Sie **Menü** -> **Struktur** -> **Rubriken**. Sie gelangen zur **Rubriken-Verwaltung**. Dort können Sie eine neue Rubrik erstellen oder eine bestehende Rubrik bearbeiten.

Der Dialog ist in folgende Abschnitte unterteilt:

- Rubriken-Information
- Workflow-Auswahl
- Metadateien und Templates auswählen
- Zugriffsrechte
- Asset-Rubriken

Rubriken-Eigenschaften können in einem Template verwendet werden. Lesen Sie hierzu Kapitel **Eigenschaften der Rubrik auslesen** im Programmierhandbuch.

5.1.1 Rubriken-Informationen angeben

Die Rubriken-Informationen dienen als Default-Einstellung für die in einer Rubrik erzeugten Dokumente. Sie können durch Workflow-Plug-Ins jederzeit während des Workflows verändert werden, so dass ein Dokument beispielsweise nicht unbedingt in dem hier angegebenen Verzeichnis gespeichert wird.

Folgende Informationen geben Sie in diesem Abschnitt des Dialogs ein:

1. Tragen Sie im Feld **Name** den Namen der Rubrik ein.

   Dieser Name erscheint in der Rubrikenliste und für jedes Dokument in der entsprechenden Spalte der Schreibtisch-Ansicht.

2. Tragen Sie im Feld **Beschreibung** eine kurze Beschreibung der Rubrik ein.

   Die Beschreibung erscheint im Dialog **Rubriken-Eigenschaften** und wird im Archiv verwendet.

3. Geben Sie das Verzeichnis an, in dem alle in dieser Rubrik erzeugten Dokumente gespeichert werden sollen.

   Dieser Wert kann, während das Dokument den Workflow durchläuft, geändert werden. Es ist nicht erlaubt, Leerzeichen in Verzeichnisnamen zu verwenden.

4. Tragen Sie im Feld **Dateiname** den Default-Dateinamen für Dokumente ein, die in dieser Rubrik erzeugt werden.

   Dieser Wert kann, während das Dokument den Workflow durchläuft, geändert werden. Es ist nicht erlaubt, Leerzeichen in Dateinamen zu verwenden.

5. Meta-Information: Klicken Sie auf den Link Bearbeiten, um zusätzliche Meta-Informationen für die Rubrik einzugeben. Diese sind dann für alle Dokumente dieser Rubrik sowie für anschließend erzeugt Unterrubriken gültig. Bereits bestehende Unterrubriken "erben" nachträglich hinzugefügte Meta-Informationen nicht! Lesen Sie hierzu auch Abschnitt 3.1.2 **Rubriken bearbeiten** auf Seite 41.

   Hinweis:

   Diese Option steht nur bei der nachträglichen Bearbeitung bestehender Rubriken zur Verfügung.

6. Soll die neue Rubrik nur zur Strukturierung des Rubrikenbaums dienen, wählen Sie die Option **nur Unterrubriken**. In einer solchen Rubrik können nur Unterrubriken, jedoch keine Dokumente angelegt werden.

   Achtung:

   Wird die Option **nur Unterrubriken** *ausgewählt, können keine Dokumente, die ursprünglich in dieser Rubrik erzeugt wurden, zurückimportiert werden. Dies greift auch dann, wenn Dokumente aus dem Archiv reimportiert werden sollen.*

   Die Option **Dokumente und Unterrubriken** erlaubt sowohl die Erzeugung von Dokumenten in dieser Rubrik als auch das Anlegen von Unterrubriken.

5.1.2 Workflow festlegen

Der Workflow, der für die Rubrik verwendet werden soll, wird im Abschnitt **Workflow festlegen** des Dialogs **Rubriken-Eigenschaften** bestimmt. Da ein Workflow mehrere Einstiegspunkte haben kann, wird nicht nur der Workflow, sondern auch der Einstiegspunkt bestimmt.

Jedes in der Rubrik erzeugte Dokument erhält eine Kopie des hier eingestellten Workflows und durchläuft alle zu diesem Workflow gehörenden Schritte, bis es den Workflow verlässt. Dokumente, die sich innerhalb des Workflows befinden, werden von Änderungen dieses Workflows nicht beeinflußt.

In der Pull-down-Liste werden alle Einstiegspunkte aller zur Verfügung stehenden Workflows in der Form

Name des Workflows –> Name des Einstiegspunkts

aufgelistet.

Wählen Sie den gewünschten Workflow-Einstiegspunkt aus.

Ein Dokument, das in der Rubrik erstellt wird, durchläuft alle Workflow-Schritte, die dem ausgewählten Workflow-Einstiegspunkt folgen.

5.1.3 Metadatei und Template bestimmen

In diesem Abschnitt werden die Metadatei und das Template bestimmt, die in der Rubrik verwendet werden sollen.

Diese Einstellungen können über eine Template-Auswahl während der Eingabe der Metainformationen jederzeit überschrieben werden. In der Pull-down-Liste **Metadatei** werden alle Metadateien aufgelistet, die sich im Verzeichnis `/site/metafiles` befinden.

Wählen Sie die Metadatei aus, die in der Rubrik verwendet werden soll.

Hinweis:

Sollte im ausgewählten Workflow kein MetaEdit-Schritt verwendet werden, muss trotzdem eine Metadatei ausgewählt werden.

In der Pull-down-Liste **Template** sind alle Templates aufgelistet, die sich im Verzeichnis */site/templates* befinden. Der Eintrag in der Liste wird aus der Templatebeschreibung und der Templatenummer (in Klammern) gebildet.

Wählen Sie das Template aus, das in der Rubrik verwendet werden soll.

Hinweis:

Sowohl Metadatei als auch Template können im Laufe des Workflows gewechselt werden. Lesen Sie hierzu auch Abschnitt 7.5.6 **Bearbeiten** *auf Seite 99, Abschnitt 7.5.23* **Metaedit** *auf Seite 116, sowie Abschnitt 7.5.24* **Metadaten setzten** *auf Seite 118.*

5.1.4 Rubriken-Zugriff

Im Folgenden wird beschrieben, wie der Zugriff auf Rubriken eingeschränkt werden kann. Das Zugriffsrecht auf eine Rubrik kann in vier verschiedenen Abstufungen vergeben werden:

- Lesen (des Rubrikinhaltes),
- Dokumente erzeugen,
- Dokumente löschen und
- Rubrik bearbeiten.

Für jede dieser Stufen können Rollen bestimmt werden, die Zugriff auf die Rubrik erhalten.

Das Recht, in einer Rubrik Dokumente erstellen zu können, beinhaltet gleichzeitig das Recht, Dokumente, die in dieser Rubrik erstellt wurden, freizuschalten. Das Recht, den Menüpunkt **Dokumente freischalten** im Menü sehen zu können, muss ebenfalls für die betreffende Rolle über die Menü-Verwaltung gesetzt sein.

5.1.4.1 Zugriffsrechte zuweisen

Zugriffsrechte für Rubriken werden im Abschnitt **Rechte bearbeiten** des Dialogs **Rubriken-Eigenschaften** eingestellt. Um einer Rolle Zugriffsrechte mit einer bestimmten Stufe zu erteilen, gehen Sie wie folgt vor:

1. Markieren Sie in dem Auswahlfeld **Verfügbare Rollen** einen oder mehrere Einträge.

   Mehrere Einträge werden mit Hilfe der Strg-Taste markiert.

2. Klicken Sie auf den zwischen den Auswahlfeldern stehenden nach rechts weisenden Pfeil-Button.

Die ausgewählten Rollen werden in die Liste **Gewählte Rollen** übertragen.

Nimmt ein Benutzer eine der ausgewählten Rollen an, so erhält er auf die Rubrik mit der entsprechenden Stufe Zugriff.

5.1.4.2 Zugriffsrecht aberkennen

Um ein zugewiesenes Zugriffsrecht wieder abzuerkennen, gehen Sie wie folgt vor:

1. Markieren Sie in dem Auswahlfeld **Gewählte Rollen** die Rollen, die nicht mehr zum Zugriff mit der entsprechenden Stufe berechtigen sollen.

   Mehrere Einträge werden mit Hilfe der Strg-Taste markiert.

2. Klicken Sie auf den zwischen den Auswahlfeldern stehenden nach links weisenden Pfeil-Button.

   Die markierten Rollen werden aus der Liste **Gewählte Rollen** entfernt.

5.1.5 Asset-Rubriken

Eine Asset-Rubrik besitzt alle Eigenschaften einer herkömmlichen Dokumenten-Rubrik in **imperia** . Sie können in den Einstellungen einer Asset-Rubrik Metainformationen hinterlegen, die für alle darin angelegten Assets verfügbar sind. Die Details zu den Asset-Rubriken finden Sie in Abschnitt 11.1 **Mit Asset-Rubriken arbeiten** auf Seite 140.

5.1.5.1 Asset-Rubriken in 'Freigeschaltete Dokumente bearbeiten' anzeigen

Im **Freigeschaltete Dokumente bearbeiten** Dialog gibt es die Möglichkeit, die Liste der angezeigten Dokumente auf bestimmte Rubriken zu beschränken. Damit in der Auswahlliste auch die gewünschte Asset-Rubrik angezeigt wird, muss folgende Metavariable in der entsprechenden Rubrik hinzugefügt werden:

- `__imperia_show_in_cat_filter` = „1“

Kapitel 6. Benutzer und Rollen

Die Benutzer-Verwaltung ermöglicht es, abhängig vom Umfang der erworbenen Lizenz Benutzer und Rollen anzulegen und zu verwalten.

Bevor Benutzer und Rollen angelegt werden empfehlen wir, ein Konzept über die zukünftige Struktur des Systems zu erstellen. Je genauer die Struktur geplant wird, desto einfacher und unkomplizierter ist nicht nur die erste Konfiguration, sondern auch die spätere Erweiterung und Pflege. Folgende Überlegungen sollten in die Struktur einfließen:

Wie ist der Arbeitsablauf innerhalb des Betriebes organisiert?

Die Organisation und die Arbeitsweise Ihres Unternehmens kann durch die Benutzer- und Rollen-Struktur in **imperia** abgebildet werden. Die hier gefundenen Zusammenhänge geben erste Hinweise auf Art und Anzahl der Rollen.

Welcher Mitarbeiter nimmt welche Aufgaben wahr?

Anhand dieser Überlegungen können für die einzelnen Mitarbeiter die Rollen bestimmt werden, die ihm zur Verfügung stehen.

Welche Rollen müssen aufgrund der beiden vorherigen Überlegungen angelegt werden?

Die hier definierten Rollen werden später in der Benutzer-Verwaltung angelegt (siehe Abschnitt 6.1 **Neue Rolle anlegen** auf Seite 73).

Welche Inhalte sollen von welchen Rollen bearbeitet werden?

Diese Überlegungen fließen bei der Vergabe der Zugriffsrechte auf die Rubriken und damit auf die in dieser Rubrik erstellten Dokumente ein. Es ist auf jeden Fall sinnvoll, Zugriffsrechte auf Rubriken restriktiv zu handhaben, um ungewollte bzw. unerlaubte Änderungen zu verhindern. Die hier definierten Rubriken-Rollen-Beziehungen werden über den Rubriken-Zugriff (siehe Abschnitt 5.1.4 **Rubriken-Zugriff** auf Seite 71) umgesetzt.

Zugriffsrechte auf bestimmte Bereiche oder Funktionen können nur Rollen zugewiesen werden. Ein Benutzer hat, mit Ausnahme des Superusers, keine Rechte, sondern erhält alle Zugriffsrechte über die aktuelle Rolle, die er gerade angenommen hat.

Rollen dienen unter anderem dazu, die gleichen Zugriffsrechte mehreren Benutzern zugänglich zu machen. Einem Benutzer können auch mehrere Rollen zugewiesen werden, die er nach Belieben wechseln kann. Er hat jedoch immer nur die Zugriffsrechte, die ihm durch seine aktuell angenommene Rolle verliehen werden.

Rollen können beliebig verschachtelt werden. In der daraus resultierenden Struktur vererbt eine Rolle, die weitere Rollen enthält, ihre Zugriffsrechte auf ihre Unterrollen.

Wir empfehlen folgendes Vorgehen:

Erstellen Sie zuerst die gewünschte Rollenstruktur und geben Sie den einzelnen Rollen die entsprechenden Rechte für die Menüs, die Rubriken und die Objekte in der Mediendatenbank. Lesen Sie hierzu folgende Abschnitte:

- Abschnitt 8.2.3 **Zugriffsrechte für Menüs einstellen** auf Seite 128,
- Abschnitt 5.1.4 **Rubriken-Zugriff** auf Seite 71 und
- Abschnitt 11.3.16 **Zugriffsrechte für Asset-Rubriken verwalten** auf Seite 151.

Jeder Benutzer kann über den Link "Rollen" in der Fußzeile jederzeit eine andere Rolle annehmen und so seine aktuellen Zugriffsrechte ändern. Lesen Sie hierzu Kapitel **Rolle annehmen** im Autorenhandbuch.

6.1 Neue Rolle anlegen

Neue Rollen werden im Dialog **Rollen-Einstellungen** angelegt. Dieser Dialog wird über den Menüpunkt **Neue Rolle anlegen** im Ordner **Benutzer und Rollen** geöffnet.

Der Dialog ist in mehrere Abschnitte gegliedert:

Obligatorische Angaben

Hier wird der Name der neuen Rolle eingetragen. Der Name ist ein Pflichtfeld, ohne den die neue Rolle nicht gespeichert werden kann.

Optionale Angaben

Hier können im Feld **Kommentar** optionale Angaben wie Bemerkungen o.ä. gemacht werden.

Rollenzugehörigkeit

In diesem Abschnitt werden die Rollenzugehörigkeiten der neuen Rolle eingestellt. Wenn Sie eine neue Rolle einer anderen Rolle zuordnen, wird die neue Rolle eine Unterrolle der ausgewählten Rolle und erbt deren Zugriffsrechte. Alle Benutzer, die die neue Rolle annehmen können, erben diese Rechte ebenfalls. Die gewünschten Rollen werden in der linken Liste markiert und durch eine Klick auf den nach rechts weisenden Pfeil-Button in die rechte Liste übertragen. Der nach links weisende Pfeil-Button entfernt markierte Einträge aus der rechten Liste wieder.

Zugriffsrechte

Dieser Abschnitt wird nur angezeigt, wenn Sie bereits neue Rollen angelegt haben. Wählen Sie hier die Rollen aus, die das Änderungsrecht für die neue Rolle bekommen sollen. Die gewünschten Rollen werden in der linken Liste markiert und durch einen Klick auf den nach rechts weisenden Pfeil-Button in die rechte Liste übertragen. Der nach links weisende Pfeil-Button entfernt markierte Rollen aus der rechten Liste. Wird die Checkbox **Alle Rollen dürfen auf diese Rolle zugreifen** markiert, dürfen alle Rollen die neue Rolle ändern.

imperia verwaltet einige Zugriffsberechtigungen intern zentral über so genannte Access Control Lists (kurz ACL). In diesem Abschnitt legen Sie die Berechtigung der betreffenden Rolle auf diese Objekte fest. Es wird jedoch nicht die aktuell eingestellte ACL-Konfiguration dargestellt. In dem Drop-Down-Menü **Konfiguration wählen** können Sie einige vordefinierte Berechtigungsprofile auswählen. Bei der Auswahl eines Profils werden die entsprechenden Rechte in der Liste **Selektierte Zugriffsrechte** angezeigt. Diese Profile können nicht verändert werden, sondern dienen nur als Vorlagen. Eine Ergänzung ist jedoch möglich. Markieren Sie dazu weitere Rechte mit gedrückter Strg-Taste.

Hinweis:

Mit dieser Einstellung legen Sie die Zugriffsrechte auf die angegebenen Objekte **global** *für die betreffende Rolle fest.*

Die vergebenen Rechte sind immer global und sollten daher sparsam gesetzt werden. Beispiele:

- Template - Lesen: diese Rolle darf alle Templates lesen
- Template - Schreiben: diese Rolle darf alle Templates bearbeiten
- Rubrik - Erzeugen: diese Rolle darf Unterrubriken in allen Rubriken erstellen
- Workflows - Schreiben : diese Rolle darf alle Workflows ändern und löschen

Eingaben werden über den Button **Okay** am unteren Rand des Dialogs gespeichert. Anschließend wird ein Hinweis angezeigt, dass die Daten gespeichert werden.

Der Button **Schließen** schließt den Dialog, ohne dass Änderungen gespeichert werden.

Das Anlegen einer neuen Rolle ist damit abgeschlossen. Um weitere Zugriffsrechte einzurichten, die der neuen Rolle verliehen werden sollen, lesen Sie folgende Kapitel:

- Abschnitt 8.2.3 **Zugriffsrechte für Menüs einstellen** auf Seite 128,
- Abschnitt 5.1.4 **Rubriken-Zugriff** auf Seite 71 und
- Abschnitt 11.3.16 **Zugriffsrechte für Asset-Rubriken verwalten** auf Seite 151.

6.2 Rollendaten bearbeiten

Das Bearbeiten einer Rolle geschieht im Dialog **Rolleneinstellungen**, der über den Menüpunkt **Rollendaten bearbeiten** im Ordner **Benutzer und Rollen** geöffnet wird.

Wird dieser Menüpunkt geöffnet, erscheint zunächst eine Liste aller bereits eingerichteten Rollen. Die zu bearbeitende Rolle wird durch einen Klick auf den Namenseintrag in dieser Liste ausgewählt.

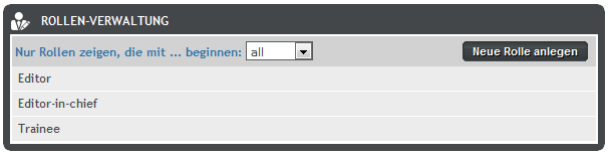


Abb. 6.1: Dialog Rollenauswahl

Der Dialog besteht aus zwei Abschnitten:

Liste einschränken

In diesem Abschnitt wird die Anzahl der im Abschnitt **Rolle** angezeigten Rollen begrenzt. Über die Drop-Down-Liste **Beginnt mit** werden die Anfangsbuchstaben der Rollennamen, die angezeigt werden sollen, eingestellt.

Rollen

In diesem Abschnitt werden nur die Rollen angezeigt, die Sie aufgrund Ihrer aktuellen Rolle sehen dürfen und deren Name mit dem im Abschnitt **Liste einschränken** eingestellten Kriterium übereinstimmen.

Die Namen der angezeigten Rollen sind verlinkt, so dass ein Klick auf den Namen den eigentlichen Dialog zum Bearbeiten der Rollendaten öffnet. Neben dem Namen der Rolle zeigt dieser Abschnitt auch die Rollen-ID.

Der Dialog zum Bearbeiten der Rollendaten ist identisch mit dem Dialog zum Anlegen einer neuen Rolle. Ein Beschreibung der einzelnen Abschnitte finden Sie in Kapitel 6.1

Der Button **Okay** am unteren Rand des Dialogs speichert die Änderungen. Es erscheint daraufhin ein Hinweis, dass die Änderungen gespeichert werden. Der Button **Schließen** schließt den Dialog, ohne Änderungen zu speichern.

6.3 Löschen einer Rolle

Wenn Sie eine vorhandene Rolle löschen, verlieren alle Benutzer, die diese Rolle annehmen können, die mit ihr verbundenen Berechtigungen.

Ein Rolle wird über den Button **Löschen** unterhalb des Dialogs **Rolleneigenschaften** gelöscht. Dieser Dialog wird über den Menüpunkt **Rollendaten bearbeiten** im Ordner **Benutzer und Rollen** geöffnet.

Wird der Button **Löschen** betätigt, erscheint eine Sicherheitsabfrage, die mit **Okay** bestätigt werden muss. Erst dann wird die ausgewählte Rolle tatsächlich gelöscht.

Achtung:

Wenn Sie nur eine begrenzte Rollenanzahl lizenziert haben, sollten Sie keine Rollen löschen, sondern bestehende editieren.

Da aus technischen Gründen einmal vergebene Rollen-IDs beim Löschen der zugehörigen Rolle nicht freigegeben werden, erhöht sich für das System beim Anlegen neuer Rollen auch dann die Gesamtanzahl der Rollen, wenn Sie vorher eine oder mehrere Rollen löschen.

6.4 Neuen Benutzer anlegen

Neue Benutzer werden im Dialog **Benutzereinstellungen** angelegt. Dieser Dialog wird über den Menüpunkt **Neuen Benutzer anlegen** im Ordner **Benutzer und Rollen** geöffnet.

Der Dialog ist in mehrere Abschnitte gegliedert:

Obligatorische Angaben

In diesem Abschnitt werden der Nachname, der Vorname, das Login und das Passwort des neuen Benutzers in die entsprechenden Felder eingetragen. Alle diese Angaben müssen getätigt werden, um den neuen Benutzer anlegen zu können. Wird ein Feld nicht ausgefüllt, erscheint beim Versuch zu speichern eine Fehlermeldung. Das gewählte Passwort muss zusätzlich durch eine identische Eingabe in das Feld **Passwort wiederholen** bestätigt werden. Gültige Passwörter umfassen mindestens fünf alphanumerische Zeichen. Passwörter mit weniger Zeichen werden vom System zurückgewiesen. Ferner wird in diesem Abschnitt die Sprache der **imperia** -Oberfläche für den neuen Benutzer eingestellt. Wird für die Sprache der Eintrag *Standard* ausgewählt, wird die **imperia** -Oberfläche in der Standard-Systemsprache angezeigt (siehe Abschnitt 2.8.7 **System-Sprache** auf Seite 27). Des Weiteren kann über eine Auswahlbox gesteuert werden, ob der Benutzer aktiv ist oder nicht.

Optionale Angaben

In diesem Abschnitt können Adressdaten oder andere Angaben zum neuen Benutzer eingegeben werden. Keines der Felder muss in diesem Abschnitt ausgefüllt werden. Diese Felder kann der Benutzer über den Menüpunkt **Persönliche Einstellungen** im Ordner **Benutzer und Rollen** selber ausfüllen. Über die Programmierung des Templates kann man auf die Werte, die in diesen Feldern gespeichert sind, zugreifen.

Rollenzugehörigkeit

In diesem Abschnitt werden die Rollen ausgewählt, die der neue Benutzer annehmen kann. Die Liste **Rollen** enthält die bereits im System angelegten Rollen. Alle Rollen, die der Benutzer annehmen darf, müssen von der linken in die rechte Liste übertragen werden. Dies geschieht über den nach rechts weisenden Pfeil-Button, der die in der linken Liste markierten Einträge in die rechte Liste überträgt. Der nach links weisende Pfeil-Button entfernt markierte Einträge in der rechten Liste wieder. Die Reihenfolge der Einträge in der rechten Liste kann über die nach oben bzw. nach unten weisenden Pfeil-Buttons verändert werden.

Zugriffsrechte

Hier wird eingestellt, welchen Rollen das Recht verliehen wird, die Daten des neuen Benutzers zu bearbeiten. Alle Rollen, den dieses Recht verliehen werden soll, müssen von der linken in die rechte Liste übertragen werden. Dies geschieht über den nach rechts weisenden Pfeil-Button, der markierte Einträge von der linken Liste in die rechte Liste überträgt. Der nach links weisende Pfeil-Button entfernt markierte Einträge aus der rechten Liste wieder. Wird die Checkbox **Alle Rollen dürfen auf diesen Benutzer zugreifen** markiert, wird allen Rollen das Recht verliehen, die Daten des neuen Benutzers zu bearbeiten.

Die Daten des neuen Benutzers werden mit Hilfe des Buttons **Okay** am unteren Rand des Dialogs gespeichert. Es wird ein Hinweis angezeigt, dass die Benutzerdaten gespeichert werden. Der Button **Schließen** schließt den Dialog, ohne Änderungen zu speichern.

6.5 Benutzerdaten bearbeiten

Das Bearbeiten von Benutzerdaten geschieht über den Dialog **Benutzereinstellungen**, der über den Menüpunkt **Benutzerdaten bearbeiten** im Ordner **Benutzer und Rollen** geöffnet wird.

Wird dieser Menüpunkt geöffnet, erscheint zunächst eine Liste aller bereits im System eingerichteten Benutzer. Der zu bearbeitende Benutzer wird über einen Klick auf den Namen ausgewählt.

BENUTZERLISTE
LISTE EINSCHRÄNKEN (Bitte schränken Sie Ihre Suche ein, wenn zu viele Benutzer angezeigt werden.)
Feldwert Nachname beginnt mit: all
Neuen Benutzer anlegen
Nachname Vorname Login E-Mail-Adresse ID
Finke Kai kfinke 3
Hoos Oliver ohoos 2
OXE oxi foo 4
superuser superuser superuser 1

Abb. 6.2: Dialog Benutzerauswahl

Der Dialog besteht aus zwei Abschnitten:

Liste einschränken

In diesem Abschnitt wird die Anzahl der im Abschnitt **Benutzer** angezeigten Einträge eingeschränkt. In der Drop-Down-Liste **Feldwert** wird das **Auswahlkriterium** ausgewählt, nach dem die Liste im Abschnitt **Benutzer** eingeschränkt werden soll. Mögliche Auswahlen sind *Name*, *Vorname*, *Login* und *Email-Adresse*. Über die Drop-Down-Liste **Beginnt mit** werden die Anfangsbuchstaben der anzuzeigenden Einträge ausgewählt. Die Liste im Abschnitt **Benutzer** wird automatisch an die Auswahl angepasst. Passt die eingestellte Einschränkung auf keinen Eintrag, bleibt die Liste leer.

Benutzer

In diesem Abschnitt werden nur die Benutzer angezeigt, die auf die Einschränkung im Abschnitt **Liste einschränken** passen. Ein Klick auf den Namen eines Benutzers öffnet den eigentlichen Dialog zum Bearbeiten der Benutzerdaten, in den die Daten des ausgewählten Benutzers eingetragen werden.

Der Dialog zum Bearbeiten von Benutzerdaten ist identisch mit dem Dialog zum Anlegen eines neuen Benutzers. Die einzelnen Abschnitte des Dialogs werden in Abschnitt 6.4 **Neuen Benutzer anlegen** auf Seite 75 beschrieben.

Änderungen an den Benutzerdaten werden über den Button **Okay** am unteren Rand des Dialogs gespeichert. Der Button **Schließen** schließt den Dialog, ohne Änderungen zu übernehmen.

6.6 Benutzer löschen

Benutzer werden über den Button **Löschen** unterhalb des Dialogs **Benutzereinstellungen** gelöscht. Dieser Dialog wird über den Menüpunkt **Benutzerdaten bearbeiten** im Ordner **Benutzer und Rollen** und über einen anschließenden Klick auf den Namen des entsprechenden Benutzers geöffnet.

Wird der Button **Löschen** betätigt, erscheint eine Sicherheitsabfrage, die mit **Okay** bestätigt werden muss. Erst daraufhin wird der Benutzer gelöscht.

Dokumente speichern die Benutzer-Id des letzten Bearbeiters. Werden Benutzer aus dem System gelöscht, so kann der Name des gelöschten Benutzers nicht mehr aufgelöst werden. Die bevorzugte Alternative zum Löschen des Benutzers ist es den Benutzer zu deaktivieren.

6.7 LDAP und Verzeichnisdienste

LDAP steht für Lightweight Directory Access Protocol und ist im Prinzip eine im Netzwerk verteilte Datenbank, die auf dem Client/Server-Prinzip basiert. Ein LDAP-Client öffnet eine Verbindung mit einem LDAP-Server und ruft entweder Informationen ab oder liefert Informationen, die in das Verzeichnis aufgenommen werden sollen. Die Art und Weise, wie dies geschieht, wird durch das Protokoll bestimmt (im Moment Version 3 von LDAP). In diesem Zusammenhang wird LDAP auch als X.500 Lite bezeichnet. X.500 ist ein internationaler Standard für Verzeichnisse, aber wesentlich komplexer aufgebaut als LDAP.

Für die Verwendung von LDAP in **imperia** muss das LDAP-Perl-Modul installiert sein. Man kann es unter `http://www.cpan.org/` oder unter `http://perl-ldap.sourceforge.net` herunterladen. In einigen Linux-Distributionen ist es enthalten, muss aber explizit installiert werden.

Da der Begriff *Datenbank* schon gefallen ist und Ähnlichkeiten auch nicht von der Hand zu weisen sind, vergleichen wir im Folgenden die Eigenschaften von LDAP mit den Eigenschaften von Datenbanken und Dateisystemen.

LDAP legt Inhalte optimiert für lesenden Zugriff ab. Datenbanken hingegen bieten beide Varianten: auf Lesezugriff optimiert (z.B.: MySQL) oder auf Schreibzugriff (Oracle). Ferner stellt LDAP keinerlei Transaktionsmechanismen zur Verfügung. Dies ist bei Datenbanken oft ein wesentlicher Sicherheitsaspekt bzgl. Datenverlusten.

Beim Dateisystem liegt der Vergleich deshalb sehr nahe, weil sich Dateisysteme zur Speicherung ihrer Dateien eines Baumes bedienen. Dies ist bei LDAP genauso.

Dateien können im Baum als ein Knoten angesehen werden und haben oft eine Länge von mehreren Giga- oder Megabytes. LDAP hingegen ist darauf ausgelegt, dass ein Knoten in seinem Verzeichnisbaum eine eher kleinere Menge an Speicher benötigt. Ein typisches Beispiel für einen Eintrag wären die Daten eines Benutzers, die in der Regel ein Megabyte nicht überschreiten dürften.

Ein anderer Punkt ist die Art des Zugriffs auf die Daten innerhalb eines Knotens. Bei Dateisystemen sind Zugriffe auf einzelne Bytes der Datei direkt möglich, d.h. ein Byte ist die kleinste lesbare und schreibbare Einheit im Dateisystem. Bei LDAP ist dies nicht möglich. Hier ist die kleinste Einheit, die man schreiben/lesen kann, das Attribut, das später erläutert wird.

imperia bietet neben der eigenen Rollen- und Benutzerverwaltung die Möglichkeit, an ein LDAP-System angebunden zu werden. Dadurch kann eine bereits vorhandene Rollen- und Benutzerdatenbank in **imperia** genutzt werden. Hierbei wird die Administration und Pflege dieses Verzeichnisses nicht mehr von **imperia** durchgeführt.

6.7.1 Einschränkungen

Bei der Einbindung eines über LDAP ansprechbaren Verzeichnisdienstes werden bestimmte Funktionen, die im Standard-Paket von **imperia** enthalten sind, eingeschränkt:

- Die Administration und Pflege der Rollen- und Benutzerinformationen kann in **imperia** nicht mehr in vollem Umfang durchgeführt werden
- Die transitive Rollenstruktur, wie im eigenen Benutzermanagement von **imperia** , wird bei der LDAP-Anbindung nicht aufgelöst. Allerdings ist dieser Punkt bei vielen Verzeichnisdiensten irrelevant, da diese selbstständig ein transitives Rollenmanagement auflösen. Die Rechte bezüglich Zugriff und Modifikation anderer Benutzer werden im Rahmen der ersten LDAP-Implementierung nicht verändert. Das heißt im Einzelnen:
  - Ein Leserecht auf alle anderen Benutzer und Rollen kann einem einzelnen Benutzer nicht aberkannt werden.
- Bei der Benutzung von ActiveDirectory, dem LDAP-Server von Microsoft®, wird zurzeit nur genau eine Windows®-Domäne unterstützt.

6.7.2 Anbindung an einen Verzeichnisdienst

Zur Anbindung an einen Verzeichnisdienst werden in **imperia** zwei Verfahren unterstützt:

imperia -Account

Dieses Verfahren gewährt dem **imperia** -Server einen eigenen Zugang zum Verzeichnisdienst, über den er die Benutzer- und Rollendaten auslesen kann. Informationen zu einem Benutzer, die **imperia** benötigt, aber nicht im Verzeichnisdienst abgelegt werden können, werden im Dateisystem des **imperia** -Servers abgelegt.

Benutzer-Account

Bei diesem Verfahren benutzt **imperia** den Zugang des jeweils eingeloggten Benutzers, um sich mit dem Verzeichnisdienst zu verbinden. Damit ist gewährleistet, dass jeder Benutzer nur die Informationen sehen kann, die ihm auf Grund seines Accounts dort auch zur Verfügung stehen. Ein Nachteil dieses Verfahrens ist, dass sich **imperia** für jeden Benutzer bei jedem Zugriff erneut einloggen muss, wofür **imperia** das Passwort des Benutzers im Cookie abspeichern muss. Dieses wird zwar verschlüsselt, es besteht aber trotzdem noch ein gewisses Sicherheitsrisiko.

6.7.3 Verfahren zur Anmeldung an imperia

Für die Anmeldung an **imperia** werden im Rahmen der LDAP-Anbindung mehrere Verfahren unterstützt:

Kerberos

Ist ein Benutzer an einem Kerberos-Server angemeldet, kann er sich mit dem Kerberos-Plug-In durch ein entsprechendes Ticket durch den Server automatisch anmelden lassen. Das Plug-In benötigt auf dem Client ein Programm, das zum Lieferumfang von **imperia** gehört. Sobald dieses Programm gestartet wird, ist er ohne Eingabe von Benutzername und Passwort in **imperia** unter seiner Identität angemeldet und wird beim Aufruf der **imperia** -Anmeldeseite automatisch ins Menü weitergeleitet. Das Kerberos-Plug-In funktioniert nur mit dem oben beschriebenen **imperia** -Account-Verfahren zur Anbindung an den Verzeichnis-Server.

LDAP

Dies ist das originäre Authentifizierungs-Plug-In für die Verzeichnisanbindung an **imperia** . Der Benutzer gibt sein Passwort für den Verzeichnisdienst bei der Anmeldung an **imperia** an. Für die weitere Arbeit mit **imperia** kann in der Konfiguration gewählt werden, ob die Anfrage von Daten mit dem Benutzer-Account oder einem einheitlichen **imperia** -Account gestartet werden.

Passwort

Dies ist das normale Authentifizierungs-Plug-In von **imperia** . Hierbei wird für die Anbindung an den Verzeichnisdienst ein **imperia** -Account vorausgesetzt. Weiterhin muss in der LDAP-Konfiguration das Passwortfeld angegeben werden. Falls dieses im LDAP-Server kodiert abgelegt wurde, muss in der LDAP-Konfiguration das Schlüsselwort `password-encryption` entsprechend gesetzt werden.

6.7.4 Notwendige Konfiguration für LDAP

Die Konfiguration der LDAP-Anbindung muss zurzeit noch in zwei Dateien manuell vorgenommen werden:

1. In der Datei `/site/config/system.conf`
2. In der Datei `/site/config/ldap.conf`

6.7.4.1 system.conf

In dieser Datei müssen folgende Variablen eingetragen werden:

```
"UMLIGHT_PLUGIN" = "LDAPLight"
"DATABASE_STORAGE_PLUGIN" = "LDAP"
```

sowie gegebenenfalls noch das verwendete Authentifizierungs-Plug-In (LDAP, Kerberos etc.):

```
"AUTH_PLUGIN" = "LDAP"
```

6.7.4.2 ldap.conf

In dieser Konfigurationsdatei für die LDAP-Anbindung müssen Informationen für den Verzeichnisdienst und Informationen über den LDAP-Zugang, mit dem **imperia** auf den Verzeichnisdienst zugreift, gespeichert werden.

Hinweis:

In Abschnitt 6.8 **Beispiele für LDAP.conf** *auf Seite 84 finden Sie LDAP.conf-Beispiele für die Gruppierungsschemata PAM und ADIR.*

6.7.4.2.1 Informationen für den Verzeichnisdienst

```
server        ldap.server.net
port          1234
version       3
```

Wird die Variable `version` nicht gesetzt, wird automatisch die Version 2 des LDAP-Protokolls verwendet.

6.7.4.2.2 Informationen über den LDAP-Zugang

Die Informationen über den LDAP-Zugang, mit dem **imperia** auf den Verzeichnisdienst zugreift, sind optional für den Fall, dass der *Anonymous Account* des Verzeichnisdienstes keine ausreichenden Leserechte besitzt, um die **imperia** -Benutzer zu identifizieren. Für spätere Zugriffe außerhalb der Authentifizierungsphase wird die LDAP-Anbindung zwei unterschiedliche Vorgehensweisen unterstützen.

```
bind_dn        cn=Admin, 0=Imperia
password       secret
```

- Liegt das Passwort des **imperia** -Benutzers vor, dann wird es verwendet, um die Benutzer- und Rolleninformationen aus dem Verzeichnisdienst auszulesen.
- Liegt das Passwort des **imperia** -Benutzers nicht vor, wird der hier definierte LDAP-Account verwendet bzw. der *Anonymous Account*, falls hier keine Angaben gemacht wurden.

67423 Liste der User-Container

Dies ist eine Liste der Container, in denen die möglichen Benutzer enthalten sind:

```
user_dn_1      ou=Entwicklung, o=Imperia, c=de
user_dn_2      ou=Verkauf, o=Imperia, c=de
user_dn_3      ou=Vertrieb, o=Imperia, c=de
```

Aus Effizienzgründen wird unterhalb der angegebenen Container die Suche nach Benutzern auf eine Ebene beschränkt. Im oben gezeigten Beispiel genügt es also nicht, folgendes anzugeben:

```
user_dn_1     o=Imperia, c=de
```

67424 Liste der Attribute

Diese Liste enthält alle Benutzer-Attribute, die im Verzeichnisdienst abgespeichert werden. Hier werden auch die Namen festgelegt, unter denen die Attribute gespeichert werden.

```
USERFIELDS
id *             uid
login *          cn
name             sn
comment          description
fname            fn
password         userPassword
telnumber        telephonenumber
END_USERFIELDS
```

Die Attribute werden als Liste zwischen den Schlüsselwörtern USERFIELDS und END_USERFIEDS eingeklammert. Die mit Sternchen gekennzeichneten Felder (`login` und `uid`) sind Pflichtfelder.

Weiterhin muss gewährleistet werden, dass die Textdarstellung des LDAP-Attributs für id nur numerisch/positiv ist und das Feld `login` innerhalb der **imperia** -Benutzer eindeutig ist.

Neben der ID sind die in **imperia** definierten Dateneingabefelder für den Benutzer als Attribute einstellbar. Diese Felder sind im Einzelnen:

```
name          comment       fname
login         password      language
homepage      addlinkmenu   addtextmenu
adddescmenu   addtexticon   bgcolor
street        city          zip
country       telnumber     cellular
faxnumber     background    email
```

Felder, die nicht im Verzeichnisdienst abgelegt werden können, werden im Dateisystem des **imperia** -Servers abgelegt.

Das Feld `password` wird dabei von **imperia** eigentlich nicht mehr benutzt, es sei denn, man verwendet anstelle des LDAP-Plug-Ins das ursprüngliche Passwort-Plug-In. In diesem Fall sollte man normalerweise auch noch die Passwort-Verschlüsselung einstellen (siehe oben).

Hinweis:

Bei dem Verzeichnisdienst von Microsoft® (ActiveDirectory) werden anstelle einer eindeutigen numerischen Benutzer- bzw. Rollen-ID die so genannten binären SIDs verwendet. Diese werden auch in **imperia** *akzeptiert, allerdings muss dafür das Gruppierungsschema auf ADIR gesetzt werden.* **imperia** *setzt im Folgenden voraus, dass die jeweiligen RIDs der SIDs innerhalb der Menge der von* **imperia** *akzeptierten Benutzer bzw. Rollen eindeutig sind. Dies ist im Falle einer einzelnen Windows® 2000 bzw. Windows® XP Domäne garantiert, andernfalls müssen Anpassungen vorgenommen werden.*

67425 Passwort-Verschlüsselung

Diese Einstellung beeinflusst nur das Standard-Authentifizierungs-Plug-In (Password):

```
password-encryption none
```

Zurzeit sind die Verfahren *none* und *MD5* implementiert.

67426 Passwort-Kodierung

Für die Anmeldung (binding) an den LDAP-Server kann eine Kodierung des Passworts vor der Übertragung notwendig sein, da es nicht im Klartext übertragen werden darf.

```
password-encoding    SASL
```

Als implementierte Standardverfahren stehen *none* und *SASL* zur Verfügung. Für die Verwendung von SASL muss Version 3 des LDAP-Protokolls verwendet werden.

67427 LDAP-Filter für gültige imperia -Benutzer

Dieser Filter dient zur Identifizierung von gültigen, also bestehenden **imperia** -Benutzern.

```
valid_user      (&(objectclass=Person) (uidnumber>=0))
```

Objekte (mit DIT), die diese erste Bedingung erfüllen, werden von **imperia** als Benutzer anerkannt.

67428 Liste der Rollen-Container

Dies ist eine Liste aller Container aus dem Verzeichnisdienst, in denen die möglichen Rollen enthalten sind:

```
group_dn_1      ou=Roles, o=Imperia, c=de
```

Hier können lediglich die Felder `id`, `name` und `comment` eingestellt werden. Das Feld `id` ist das einzige Pflichtfeld und muss in seiner Darstellung numerisch sein.

Hinweis:

Bei der Verwendung des Gruppierungsschemas ADIR gilt hierbei - wie bei den ID-Nummern der Benutzer - die gleiche Ausnahme.

67429 Liste der Rollen-Attribute

Dies ist eine Liste aller Attribute der Rolle, die im Verzeichnisdienst gespeichert werden. Diese Liste muss analog zur Liste der Benutzer in Tags geklammert werden:

```
GROUPFIELDS
id           gid
name         name
comment      description
END_GROUPFIELDS
```

674210 LDAP-Filter zur Rollen-Identifikation

Dieser Filter dient zur Identifizierung der von **imperia** verwalteten Rolle.

```
valid_group       (&(objectclass=Group)(gidnumber>=0))
```

Objekte, die diese Bedingung erfüllen, werden von **imperia** als Rollen angesehen.

674211 Gruppierungs-Schema

Welcher Benutzer welche Rolle annehmen kann und welche Rolle von welchen Benutzern angenommen werden kann, muss von **imperia** auf irgendeine Weise einsehbar sein. Bei Verzeichnisdiensten gibt es keinen einheitlichen Standard. **imperia** unterstützt zunächst das PAM-Schema.

```
grp_scheme     PAM
```

Rollen beinhalten hierbei im Attribut `memberuid` (dieses Attribut ist *multivalued*) die `uid` der Benutzer, die ihnen angehören (**Achtung!** Nicht die `uidnumber`.). Benutzer hingegen besitzen kein Attribut, in dem alle Rollen aufgeführt sind, die sie annehmen können. Im Benutzer-Datensatz erscheint unter `gidnumber` die Hauptgruppe des Benutzers, die zu den unter `memberuid` angegebenen hinzugefügt werden muss.

In einem späteren Schritt wird das Gruppierungsschema der Standard-Installation von ActiveDirectory (Windows® 2000) implementiert, die den Namen ADIR erhalten wird.

674212 Superuser-Rolle

Die Superuser-Rolle aus **imperia** (standardmäßig die Rolle mit der ID `0`) kann bei Benutzung des LDAP-Plug-Ins zur Benutzerhaltung einer beliebigen Rolle zugewiesen werden.

```
superuser_group       cn=cvs, ou=Roles, o=Imperia, c=de
```

Meldet sich ein Benutzer, der diese Rolle inne hat an, erhält er alle Rechte der Superuser-Rolle.

674213 Superuser-Zugang

Hinweis:

Diese Einstellung beeinflusst nur das LDAP-Authentifizierungs-Plug-In.

Die Verwaltung des in **imperia** angelegten Superusers kann nicht über einen im Verzeichnisdienst abgelegten Benutzer erfolgen. Trotzdem muss der Superuser auf einen im Verzeichnisdienst abgelegten Benutzer abgebildet werden. Dies geschieht mit Hilfe des folgenden Eintrags:

```
superuser       cn=tgier, ou=People, o=Imperia, c=de
```

Meldet sich ein Benutzer als Superuser an, wird das eingegebene Passwort mit den Zugangsdaten des angegebenen DN-Eintrags verglichen und ein temporärer Benutzer mit der ID `1` angelegt.

6.7.4.2.4 Benutzeranbindung

Hinweis:

Diese Einstellung beeinflusst nur das LDAP-Authentifizierungs-Plug-In.

Ist diese Variable gesetzt, wird im LDAP-Authentifizierungs-Plug-In bei jedem CGI-Aufruf eine Anbindung des Benutzers an den Verzeichnisdienst vorgenommen.

```
userbind      yes
```

Zum Verständnis: Wird diese Variable weggelassen oder auf `0` gesetzt, werden von **imperia** alle Informationen mit Hilfe des konfigurierten Administrator-Zugangs aus dem Verzeichnisdienst abgeholt.

6.7.4.2.5 Schreibzugriff

Der Schreibzugriff auf den Verzeichnisdienst via LDAP wird durch Setzen folgender Variablen unterbunden:

```
readonly      yes
```

6.7.4.3 imperia als Frontend für einen LDAP-Verzeichnisdienst

Hinweis:

Diese Funktion ist zurzeit noch in der Vorbereitung.

Soll **imperia** als Frontend für einen LDAP-kompatiblen Verzeichnisdienst verwendet werden, sind folgende Einstellungen notwendig:

6.7.4.3.1 RDN-Attribut eines neuen Benutzer-Eintrags

```
user_rdn_attribute       cn
```

6.7.4.3.2 Objektklassen eines neuen imperia -Benutzers

```
OBJECTCLASSES_USER
top
person
ImperiaUser
END_OBJECTCLASSES_USER
```

6.7.4.3.3 Das RDN-Attribut eines neuen Rollen-Eintrags

```
group_rdn_attribute      cn
```

6.7.4.3.4 Objektklassen einer neuen imperia -Rolle

Objektklassen, zu denen eine neue von **imperia** angelegte Gruppe gehört:

```
OBJECTCLASSES_GROUP
top
posixGroup
END_OBJECTCLASSES_GROUP
```

6.8 Beispiele für LDAP.conf

Im Folgenden finden Sie jeweils ein Beispiel einer `LDAP.conf` für die beiden Gruppierungsschemata PAM und ADIR.

6.8.1 PAM

```
# Servername and IP address for LDAP server.
serverldap.imperia.net
port389

# DN and password from ldap adminstrator
# None if we are using the Anonymous Binding.
#bind_dn "cn=Admin, o=intern"
#password superuser

# Container for user entries and attribute for RDN. So the DN of every
# user have the form
# "user_rdn_attribute=value_of_user_rdn_attribute, user_dn"
user_dn_1 ou=People, o=Imperia, c=de
#user_dn_2 ou=Verkauf, o=intern
#user_dn_3 ou=Einkauf, o=intern
#user_dn_4 ou=Vertrieb, o=intern
#user_dn_5 "ou=Vertrieb, o=intern"
user_rdn_attribute cn

# Timelimit for search operations
timelimit 0

# Setting attributes for imperia
 user.
USERFIELDS
id uidnumber
name cn
login uid
END_USERFIELDS
valid_user (gidnumber=100)

# Setting objectclasses an imperia
 user belongs to.
#OBJECTCLASSES_USER
#top
#account
#posixAccount
#shadowAccount
#person
#organizationalPerson
#inetOrgPerson
#END_OBJECTCLASSES_USER

# Setting attribute for imperia
 groups
GROUPFIELDS
id gidnumber
name cn
comment cn
END_GROUPFIELDS
```

```
#
group_dn_1 ou=Groups, o=Imperia, c=de

valid_group (gidnumber>=0)

# Setting objectclasses an imperia
 group entry belongs to.
#OBJECTCLASSES_GROUP
#top
#posixGroup
#END_OBJECTCLASSES_GROUP

grp_schemePAM
superuseruid=tomg, ou=People, o=Imperia, c=de
```

6.8.2 ADIR

```
# Servername and IP adress for LDAP server.
serverwindb
port389

# DN and password from ldap adminstrator
# None if we are using the Anonymous Binding.
bind_dn Administrator@activetest.hyper
password xxxxxxx

# Container for user entries and attribute for RDN.
user_dn_1 CN=Users,DC=activeTest,DC=hyper
user_rdn_attribute cn

# Timelimit for search operations
timelimit 0

# Setting attributes for imperia
 user.
USERFIELDS
id objectSid
name displayName
login cn
END_USERFIELDS

valid_user (objectclass=user)

# Setting objectclasses an imperia
 user belongs to.
#OBJECTCLASSES_USER
#END_OBJECTCLASSES_USER
# Setting attribute for imperia
 groups
GROUPFIELDS
id objectSid
name cn
comment description
END_GROUPFIELDS

#
group_dn_1 CN=Users,DC=activeTest,DC=hyper
group_dn_2 CN=Builtin,DC=activeTest,DC=hyper
valid_group (objectclass=group)

# Setting objectclasses an imperia
 group entry belongs to.
#OBJECTCLASSES_GROUP
#END_OBJECTCLASSES_GROUP
```

```
grp_schemeADIR
superuserCN=Administrator,CN=Users,DC=activeTest,DC=hyper
superuser_group CN=Domänen-Admins,CN=Users,DC=activeTest,DC=hyper
#userbind1
```

6.9 Authentifizierung über HTTP

Eine Alternative zur Benutzer-Authentifizierung über **imperia** oder einen Verzeichnisdienst bietet das Basic-Plug-In. Es unterscheidet sich von den anderen Authentifizierungsmethoden in zweierlei Hinsicht:

- Die Authentifizierung geschieht ohne Cookie.
- Die **imperia** -Authentifizierung wird außer Kraft gesetzt.

Stattdessen übernimmt der Webserver die Authentifizierung der Benutzer. Erfolgreich angemeldete Benutzer erhalten Zugriff auf das CMS.

Wichtig:

Passwörter können bei der HTTP-Authentifizierung nicht über den Menüpunkt **Persönliche Einstellungen** *geändert werden.*

Die Autorisierung angemeldeter Benutzer geschieht jedoch weiterhin über die Benutzer- und Rollenverwaltung des Systems. Ein Benutzer muss also in **imperia** existieren, damit er das System den für Ihn verfügbaren Rollen entsprechend voll nutzen kann.

Ohne **imperia** -Benutzerprofil hat ein authentifizierter Benutzer keine Rechte im System. Er hat dann lediglich Zugriff auf den Menüpunkt "Persönliche Einstellungen" und das interne Mail-System.

6.9.1 Konfiguration

Um die HTTP-Authentifizierung nutzen zu können, müssen Sie folgende Schritte ausführen:

1. Richten Sie das Verzeichnis `cgi-bin` in der Konfiguration Ihres Webservers als zugriffsgeschützten Bereich ein. Die hierfür notwendigen Schritte entnehmen Sie bitte der Dokumentation Ihres Webservers.

   Hinweis:

   Beachten Sie, dass **imperia** *die Autorisierung anhand der Benutzerdaten vornimmt, die vom Webserver übergeben werden. HTTP-Authentifizierungsdaten und* **imperia** *-Benutzerprofile müssen also übereinstimmen, damit den Benutzern ihre jeweiligen Rechte korrekt zugewiesen werden können.*

2. Die normale **imperia** -Login-Seite (`htdocs/imperia/index.html`) sollte bei der HTTP-Authentifizierung umgangen werden. Diese Seite sollte also nicht mehr erreichbar sein. Stattdessen sollte direkt das Skript `site_main.pl` im Verzeichnis cgi-bin aufgerufen werden. Wenn Sie von anderen Seiten aus auf das CMS verweisen, müssen Sie diese Verweise bei der Umstellung auf die Authentifizierung per HTTP entsprechend anpassen. Sie können beispielsweise ein HTML-Dokument mit einem Verweis auf das Skript `site_main.pl` im cgi-bin-Verzeichnis erstellen. Benennen Sie die Datei entsprechend, dass sie automatisch aufgerufen wird, wenn jemand auf den Produktionsserver zugreift. Legen Sie dieses Dokument im Document-Root Ihres Produktionssystems ab.
3. Aktivieren Sie die HTTP-Authentifizierung in **imperia** . Suchen Sie hierzu in der Datei `system.conf` im Verzeichnis `site/config` die Variable `AUTH_PLUGIN` und ändern Sie diese folgendermaßen:

   ```
   "AUTH_PLUGIN" = "Basic"
   ```

Wichtig:

Es ist auch möglich, die HTTP-Authentifizierung des Webservers dem herkömmlichen **imperia** *-Login vorzuschalten. Diese Variante ist aber wenig sinnvoll, da Benutzer sich dann zusätzlich über die* **imperia** *-Authentifizierung anmelden müssen. Ist dann das cgi-bin-Verzeichnis nicht passwortgeschützt, ist die Anmeldung am System sogar ganz unmöglich, da die Daten aus dem HTTP-Header nicht mit dem Webserver abgeglichen werden können.*

Bei einem Zugriff auf site_main.pl erscheint nun die Anmeldeaufforderung des Webservers. Nach erfolgter Authentifizierung wird das Menü von **imperia** angezeigt.

Kapitel 7. Workflows

imperia bietet die Möglichkeit, den Produktionsprozess eines Dokuments an die bestehenden firmeninternen Prozesse anzupassen. Jeder einzelne Schritt im Arbeitsablauf, beispielsweise Anlegen, Editieren, Prüfen, und Freischalten eines Dokuments, wird durch einen Workflow-Schritt im **imperia** -Workflow repräsentiert.

Aufbau eines Workflows

Ein Workflow besteht aus einem oder mehreren Workflow-Schritten, von denen jeder eine bestimmte Operation durchführt. Nur dazu autorisierte Benutzer können einen solchen Schritt ausführen. Alle benötigten Informationen zu einem Dokument sollten gesammelt werden, während es den Workflow durchläuft.

Denkbar wäre beispielsweise folgender Workflow: Im ersten Schritt werden die Meta-Informationen zu einem neuen Dokument eingegeben. Danach könnte man die Eingabe von Inhalten in dieses Dokument durch einen Autor als Workflow-Schritt definieren. Ein weiterer Schritt könnte eine Prüfung des Dokuments durch eine Prüfinstanz (Chefredakteur oder Rechtsabteilung) enthalten, in einem anderen könnte eine weitere Prüfung stattfinden (Mehr-Augen-Prinzip). Daran anschließen könnte sich ein Schritt, in dem wahlweise die Konvertierung des Dokuments in ein anderes Format (PDF, XML etc.) oder bei Eintreffen bestimmter Bedingungen dessen Löschung erfolgt.

Jeder Workflow hat mindestens einen Einstiegspunkt. Alternativ sind auch mehrere Einstiegspunkte möglich, so dass Dokumente aus verschiedenen Rubriken in einem Workflow unterschiedliche Wege durchlaufen können. Im Verlauf des Workflows können diese alternativen Abläufe zum Beispiel in einer Prüfinstanz wieder zusammenlaufen. Welchen der verfügbaren Einstiegspunkte ein Dokument nimmt, legen Sie bei der Erstellung von Rubriken fest.

Die einzelnen Schritte eines Workflows sind untereinander verbunden, wodurch sich jeder mögliche Weg, den ein Dokument darin gehen kann, abbilden lässt. Dazu gehören auch Verbindungen, die rückwärts führen, zum Beispiel dann, wenn die prüfende Instanz ein Dokument ablehnt, weil es fehlerhaft oder unvollständig ist. In so einem Fall geht das Dokument zur Korrektur zurück an eine bearbeitende Instanz.

Dokumente werden im Workflow völlig unabhängig voneinander behandelt, so dass mehrere Dokumente gleichzeitig den selben Schritt durchlaufen können.

Aufbau eines Workflow-Schrittes

Jeder Workflow-Schritt wird durch ein entsprechendes Workflow-PlugIn realisiert. Ihnen stehen eine Reihe von Standard-PlugIns zur Verfügung. Individuell angepasste PlugIns können Sie bei Bedarf erwerben oder entwickeln lassen.

Jeder Workflow-Schritt hat mindestens je einen Dateneingang sowie Datenausgang. Der Ausgang eines Workflow-Schritts wird mit einem Eingang des darauf folgenden Workflow-Schritts verbunden, woraus sich der Weg für die Dokumente durch den Workflow ergibt.

Ein Datenausgang, der nicht mit dem Eingang eines anderen Workflow-Schrittes verbunden ist, fungiert automatisch als Ende des Workflows bzw. Workflow-Zweigs. Dokumente, die einen solchen Schritt durchlaufen, verlassen anschließend den Workflow.

Vom Datenausgang eines Workflow-Schritts kann jedoch immer nur ein Weg weiterführen. Mehrere Ausgänge können durch Verwendung einer Weiche, an der das Dokument dupliziert wird, verwirklicht werden.

Änderungen am Workflow

Änderungen an einem Workflow betreffen nur Dokumente, die neu erzeugt oder aus dem Archiv re-importiert werden. Dokumente, die sich bereits vor der Änderung im Workflow befunden haben, sind nicht von den Änderungen betroffen.

Wenn Sie an Workflows für den Hermes-Daemon Änderungen vornehmen, ist ein Neustart des Daemons notwendig, um die Änderungen zu aktivieren. Dies trifft beispielsweise für automatisierte Import-Workflows zu.

7.1 Workflows verwalten

Sämtliche Funktionen für die Verwaltung von Workflows sind im Menü **Workflow** zusammengefasst. Hier können Sie Workflows neu angelegen, ändern und löschen. Ferner lassen sich Zugriffsrechte für jeden Workflow setzen oder entziehen. Im Einzelnen sind das folgende Rechte:

- Schreiben
- Ändern
- Lesen und
- Löschen

Zum Ändern von Dateirechten lesen Sie bitte auch Abschnitt 3.11 **Dateirechte in imperia verwalten** auf Seite 60.

7.1.1 Workflow-Zugriffsrechte ändern

Workflow-Zugriffsrechte werden über den Menüpunkt **Ändern der Workflow-Zugriffsrechte** im Ordner **Struktur** geändert. Es öffnet sich folgender Dialog:

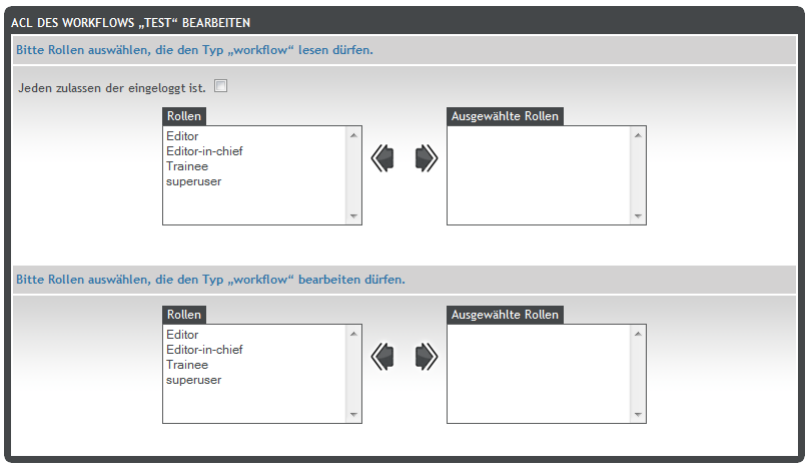


Abb. 7.1: Dialog Ändern der Workflow-Zugriffsrechte

Der Dialog ist in zwei Abschnitte gegliedert:

Dateiliste

Diese Liste enthält alle vorhandenen Workflows. Zu jedem Eintrag gibt es eine Checkbox, über die der entsprechende Workflow zum Ändern markiert wird. Die Checkbox in der blauen Titelleiste markiert gleichzeitig alle Einträge in der Liste.

Rechte

In diesem Abschnitt werden Schreib-, Lese- und Löschrechte für die in der Dateiliste markierten Workflows gesetzt. Lesen Sie hierzu Abschnitt 3.11 **Dateirechte in imperia verwalten** auf Seite 60.

7.1.2 Neuen Workflow anlegen

Um einen Workflow anzulegen, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Workflow anlegen** im Ordner **Workflow**. Es wird folgender Dialog geöffnet:

Abb. 7.2: Dialog Workflow anlegen

Tragen Sie den Namen des neuen Workflows ein. Der Name darf nur aus Buchstaben, Ziffern, dem Minus-Zeichen und dem Unterstrich (_) bestehen.

2. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

   Es wird das Grid gestartet, in dem der neue Workflow gestaltet und gespeichert wird. Lesen Sie hierzu auch Abschnitt 7.2.3 **Bedienung des Grids** auf Seite 91.

7.1.3 Workflow bearbeiten

Workflows werden im Grid bearbeitet. Gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Workflow bearbeiten** im Ordner **Workflow**. Es wird eine Liste aller vorhandenen Workflows angezeigt.
2. Klicken Sie auf den Button **Bearbeiten** neben dem entsprechenden Workflow.

   Es wird das Grid gestartet, in dem der gewählte Workflow bearbeitet werden kann.

   Die Bedienung des Grids entnehmen Sie bitte Kapitel 7 **Workflows**.

7.1.4 Workflow löschen

Um einen Workflow zu löschen, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Workflow löschen** im Ordner **Workflow**. Es wird eine Liste aller vorhandenen Workflows angezeigt.
2. Klicken Sie auf den Button **Löschen** neben dem entsprechenden Workflow.

   Es erscheint eine Sicherheitsabfrage, die Sie mit **Okay** bestätigen müssen. Anschließend wird der Workflow gelöscht. Der Button **Zurück** bricht die Aktion ab.

7.2 Das Grid

Beim Grid handelt es sich um eine Java-Applikation, mit deren Hilfe Workflows innerhalb von **imperia** erstellt und verwaltet werden. Um mit dem Grid arbeiten zu können, muss auf dem Client-Rechner für den benutzten Browser das Java-PlugIn installiert werden.

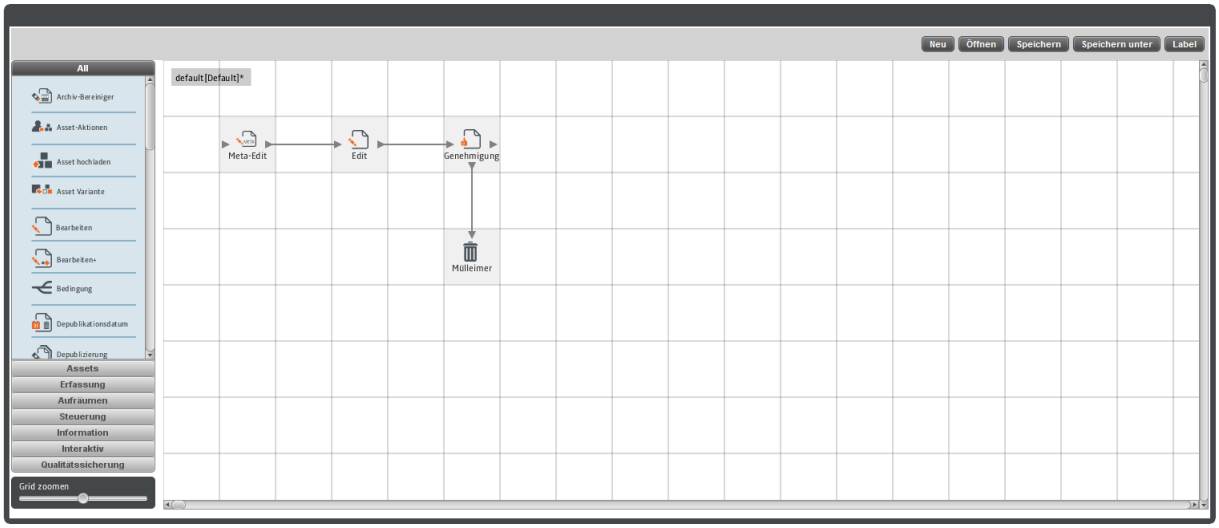


Abb. 7.3: Das Grid

Die Bedienungsoberfläche des Grids gliedert sich in folgende Bestandteile:

- PlugIn-Leiste,
- Button-Leiste und

- Arbeitsbereich.

7.2.1 PlugIn-Leiste

In der PlugIn-Leiste werden alle verfügbaren PlugIns durch Icons in einem Akkordion dargestellt. Die PlugIns werden gruppiert nach Bereichen angezeigt.

7.2.2 Button-Leiste

In der Button-Leiste stehen verschiedene Buttons für Dateioperationen zur Verfügung. Weiterhin wird am rechten Rand der Titel des aktuell geladenen Workflows angezeigt. Die Buttons haben im einzelnen folgende Funktion:

Speichern

Speichert den geladenen Workflow und übernimmt Änderungen.

Speichern unter

Speichert den geladenen Workflow unter einem neuen Namen.

Öffnen

Öffnet die Liste aller eingerichteten Workflows, aus der dann ein Workflow zum Öffnen ausgewählt werden kann. Lesen Sie hierzu Abschnitt 7.1.3 **Workflow bearbeiten** auf Seite 90.

Neu

Legt einen neuen Workflow an. Es öffnet sich der Dialog **Workflow anlegen**. Lesen Sie hierzu Abschnitt 7.1.2 **Neuen Workflow anlegen** auf Seite 89.

Titel

Öffnet einen Dialog, in dem Titel und Beschreibung des aktuellen Workflow geändert werden können.

Info

Öffnet einen Dialog, in dem Copyright- und Versionsinformationen angezeigt werden.

7.2.3 Bedienung des Grids

Zur Bedienung des Grids wird vor allen Dingen die Maus verwendet. Eine Bedienung ausschließlich über die Tastatur ist nicht möglich.

Neue Workflow-Schritte binden Sie in den Workflow ein, indem Sie das entsprechende Icon aus der PlugIn-Leiste per Drag-and-Drop im Arbeitsbereich des Grids positionieren. Lesen Sie hierzu Abschnitt 7.2.3.2 **Workflow-Schritt anlegen** auf Seite 92. Jedes Feld des Arbeitsbereichs kann genau einen Workflow-Schritt aufnehmen. Bereits vorhandene Workflow-Schritte markieren Sie mit einem Klick der linken Maustaste. Markierte Workflow-Schritte sind orange unterlegt. Das gleichzeitige Markieren mehrerer Felder ist nicht möglich.

Bei konfigurierbaren Workflow-Schritten öffnen Sie per Doppelklick einen Dialog, in dem Sie die festgelegten Eigenschaften zu diesem Workflow-Schritt konfigurieren können. Lesen Sie hierzu auch Abschnitt 7.2.3.3 **Eigenschaften eines Workflow-Schrittes einstellen** auf Seite 92. Ein Rechtsklick auf einen beliebigen im Arbeitsbereich des Grids positionierten Workflow-Schritt öffnet das Kontextmenü mit den Befehlen **Bearbeiten**, **Duplizieren** und **Löschen**.

7.2.3.1 Workflow-Schritt

Workflow-Schritte werden durch die Workflow-PlugIns repräsentiert, die Sie in der PlugIn-Leiste des Grids sehen.

Bei den Workflow-Schritten unterscheidet man grundsätzlich zwischen interaktiven und nicht interaktiven PlugIns. Interaktive PlugIns benötigen beim Durchlauf des Workflows eine Eingabe des Benutzers. Ein Beispiel hierfür ist das Bearbeiten-PlugIn. Nicht interaktive PlugIns laufen automatisiert ab und benötigen daher keine Eingabe durch den Benutzer. Ein Beispiel hierfür ist der AutoPublisher. Zur besseren Unterscheidung sind die Icons interaktiver Workflow-PlugIns mit zwei Figuren kenntlich gemacht.

Außerdem sind einige PlugIns konfigurierbar bzw. müssen Sie ihre Konfiguration auf den jeweiligen Workflow anpassen, damit Sie sie verwenden können. Bei den interaktiven PlugIns handelt es sich hierbei meistens um die Einstellung der Zugriffsberechtigungen. Nicht interaktive PlugIns haben sehr unterschiedliche Einstellungsmöglichkeiten. Lesen Sie hierzu die Beschreibung des jeweiligen PlugIns in Abschnitt 7.5 **Workflow-PlugIns** auf Seite 96.

7.2.3.2 Workflow-Schritt anlegen

Um einen Workflow-Schritt dem Workflow hinzuzufügen, wird eine Instanz eines PlugIns im Arbeitsbereich erstellt:

- Ziehen Sie per Drag & Drop das PlugIn, das Sie verwenden wollen, aus der PlugIn-Leiste in den Arbeitsbereich und positionieren Sie es in einem freien Feld. Es wird eine Instanz des PlugIns, und somit ein Workflow-Schritt, erzeugt.
- Anschließend müssen Sie das neue PlugIn mit anderen Workflow-Schritten verbinden. Lesen Sie hierzu Abschnitt 7.2.3.7 **Workflow-Schritte verbinden** auf Seite 94.
- Einige PlugIns müssen außerdem noch konfiguriert werden. Lesen Sie hierzu den folgenden Abschnitt sowie die Beschreibung der einzelnen PlugIns in Abschnitt 7.5 **Workflow-PlugIns** auf Seite 96.

Je nach Aufbau eines Workflows kann ein PlugIn mehrfach darin vorkommen.

7.2.3.3 Eigenschaften eines Workflow-Schrittes einstellen

Um die Eigenschaften eines Workflow-Schrittes einzustellen, gehen Sie wie folgt vor:

- Doppelklicken Sie im Arbeitsbereich des Grids auf den Workflow-Schritt, dessen Eigenschaften Sie einstellen wollen.

  Es wird der Dialog **Workflow-Schritt bearbeiten** geöffnet. Dieser Dialog kann auch über das Kontext-Menü der rechten Maustaste geöffnet werden (Option **Bearbeiten**). Die Einstellungsmöglichkeiten in dem Eigenschaften-Dialog sind für die einzelnen PlugIns unterschiedlich. Die folgende Abbildung zeigt exemplarisch das Dialogfenster für das Edit-PlugIn:

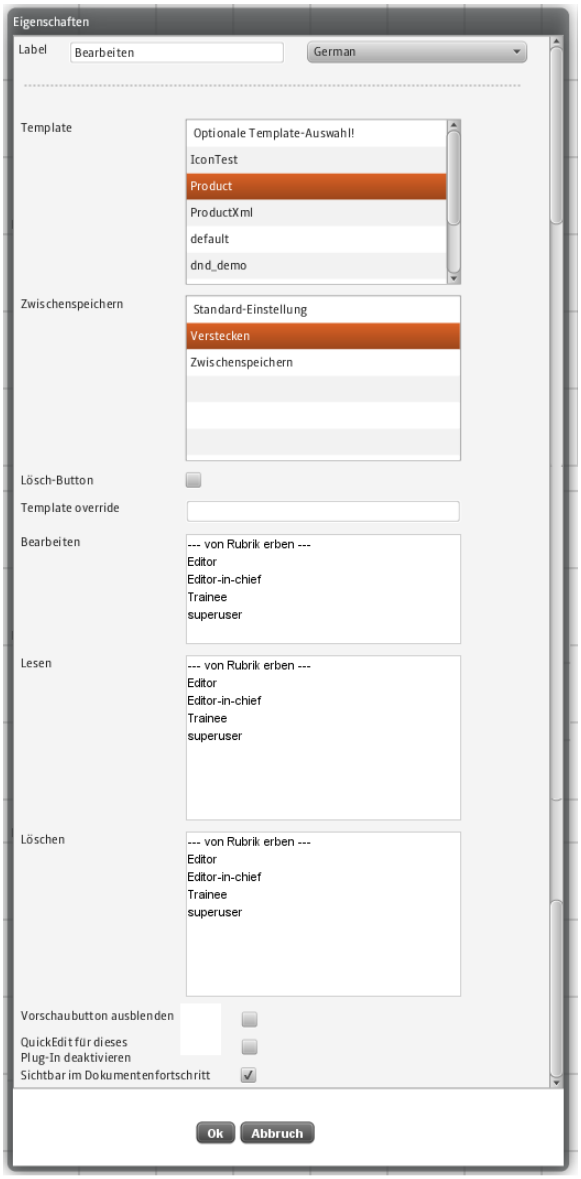


Abb. 7.4: Dialog Workflow-Schritt Eigenschaften (Edit-PlugIn)

Einige PlugIns benötigen keine weitere Konfiguration. Lesen Sie mehr über die Konfigurationsmöglichkeiten der einzelnen PlugIns im Abschnitt 7.5 **Workflow-PlugIns** auf Seite 96

7.2.3.4 Workflow-Schritt im Arbeitsbereich verschieben

Workflow-Schritte werden innerhalb des Arbeitsbereichs mit der Maus verschoben. Klicken Sie den entsprechenden Workflow-Schritt an, halten Sie die Maustaste gedrückt und verschieben Sie den Workflow-Schritt in ein anderes freies Feld. Bereits existierende Verbindungen bleiben erhalten.

7.2.3.5 Workflow-Schritt duplizieren

Wird beim Verschieben gleichzeitig die Strg-Taste gedrückt gehalten, wird der markierte Workflow-Schritt dupliziert. Er kann dann beliebig innerhalb des Arbeitsbereiches verschoben und neu positioniert werden. Eventuell vorhandene Verbindungen des ursprünglichen Workflow-Schrittes werden nicht dupliziert.

7.2.3.6 Workflow-Schritt löschen

Markierte Workflow-Schritte werden mit Hilfe der Entf-Taste oder mit Hilfe des Kontextmenüs der rechten Maustaste gelöscht. Wird ein Workflow-Schritt gelöscht, werden auch die bestehenden Verbindungen zu anderen Workflow-Schritten entfernt.

7.2.3.7 Workflow-Schritte verbinden

Das Verbinden der einzelnen Workflow-Schritte untereinander erzeugt die Wege, die ein Dokument im Workflow gehen kann. Workflow-Schritte werden mit Hilfe der Maus untereinander verbunden. Ausgehend vom Datenausgang eines Workflow-Schrittes wird mit gedrückter linker Maustaste eine Verbindung, symbolisiert durch eine schwarze Linie, zu einem Dateneingang des folgenden Workflow-Schrittes gezogen. Wird die Maustaste an einem Dateneingang losgelassen, ist die Verbindung etabliert.

7.2.3.8 Verbindungen lösen

Falls eine falsche Verbindung etabliert wurde, kann diese entweder gelöscht oder durch eine korrekte Verbindung ersetzt werden.

Zum Löschen klickt man mit der rechten Maustaste auf die Verbindung und wählt im Kontext-Menü den Eintrag **Diese Verbindung löschen**.

Um eine Verbindung zu ersetzen, muss eine neue Verbindung ausgehend vom gleichen Datenausgang, den auch die falsche Verbindung verwendet, zum korrekten Dateneingang gezogen werden. Beim Loslassen der Maustaste wird die alte Verbindung durch die neue ersetzt.

7.2.3.9 Arbeitsblatt bewegen

Das gesamte Arbeitsblattes kann bewegt werden, in dem man die rechte Maustaste gedrückt hällt und die Maus bewegt.

7.2.4 Zugriffsrechte für Workflow-Schritte

Für viele Workflow-Schritte kann festgelegt werden, welche Rollen zum Zugriff auf diesen Schritt berechtigen.

Alle Benutzer, die eine der Rollen annehmen können, erhalten dann über ihren Schreibtisch Zugriff auf die Dokumente, für die der jeweilige Workflow-Schritt ausgeführt werden soll, und zwar dann, wenn das Dokument diesen Workflow-Schritt im Workflow erreicht.

Auf den Schreibtisch der Benutzer, die diesen Schritt nicht ausführen dürfen, werden diese Dokumente nicht angezeigt. Für jeden Workflow-Schritt muss mindestens eine Rolle zum Zugriff berechtigen. Andernfalls laufen die Dokumente an diesem Workflow-Schritt auf, ohne dass sie bearbeitet werden können.

In der Praxis sieht ein Benutzer, der aufgrund seiner aktuellen Rolle Inhalt in Dokumente eingeben darf, ein Dokument also erst dann, wenn der vorangegangene Workflow-Schritt, zum Beispiel Meta-Informationen eingeben, abgeschlossen ist und das Dokument den nächsten Workflow-Schritt, zum Beispiel Inhalt eingeben, erreicht.

Hat der Benutzer nur Zugriff auf den Workflow-Schritt Inhalt eingeben, erscheint das Dokument, sobald er diesen Workflow-Schritt abschließt, nicht mehr auf seinem Schreibtisch. Es erscheint dann auf dem Schreibtisch der Benutzer, die den nächsten Workflow-Schritt, zum Beispiel Prüfung, ausführen dürfen.

7.2.4.1 Zugriffsrechte einstellen

Um die Zugriffsrechte für einen Workflow-Schritt einzustellen, gehen Sie wie folgt vor:

1. Klicken Sie im Arbeitsbereich des Grids doppelt auf den Workflow-Schritt, dessen Eigenschaften Sie einstellen wollen.

   Es wird der Dialog **Workflow-Schritt bearbeiten** geöffnet. Dieser Dialog kann auch über das Kontext-Menü der rechten Maustaste geöffnet werden.

   Es gibt drei Stufen von Zugriffsrechten auf einen Arbeitsschritt:

   - Bearbeiten
   - Lesen
   - Löschen

   Bearbeiten

   Eine Rolle, die Zugriff auf einen Workflow-Schritt mit der Stufe **Bearbeiten** erlaubt, berechtigt zum Bearbeiten von Dokumenten, die diesen Workflow-Schritt erreichen.

Lesen

Eine Rolle, die Zugriff auf einen Workflow-Schritt mit der Stufe **Lesen** erlaubt, berechtigt zum Lesen von Dokumenten, die diesen Workflow-Schritt erreichen.

Löschen

Eine Rolle, die Zugriff auf einen Workflow-Schritt mit der Stufe **Löschen** erlaubt, berechtigt zum Löschen von Dokumenten, die diesen Workflow-Schritt erreichen.

2. Markieren Sie in den entsprechenden Abschnitten eine oder mehrere Rollen, die auf den gewählten Workflow-Schritt Zugriff der entsprechenden Stufe bekommen sollen.

   Mehrere Rollen werden mit Hilfe der Strg-Taste markiert.

   Über den Eintrag **von Rubrik erben** werden die Rechte aus der mit diesem Workflow verknüpften Rubrik übernommen. Die in der Rubrik für einzelne Rollen konfigurierten Lese-, Schreib- und Löschrechte gelten dann auch für das entsprechende Workflow-PlugIn.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um die Einstellungen zu speichern. Der Button **Abbrechen** schließt den Dialog, ohne Änderungen zu speichern.

7.2.4.2 Zugriffsrechte entfernen

Um Zugriffsrechte auf einen Workflow-Schritt zu entfernen, gehen Sie wie folgt vor:

1. Klicken Sie im Arbeitsbereich des Grids doppelt auf den Workflow-Schritt, dessen Eigenschaften Sie einstellen wollen.

   Es wird der Dialog **Workflow-Schritt bearbeiten** geöffnet. Dieser Dialog kann auch über das Kontext-Menü der rechten Maustaste geöffnet werden.

2. Entfernen Sie in den entsprechenden Abschnitten die Markierung der Rollen, denen in Zukunft kein Zugriff auf den gewählten Workflow-Schritt gestatten werden soll.

   Klicken Sie dazu mit gedrückter Strg-Taste auf einen markierten Eintrag. Die Markierung wird daraufhin aufgehoben und die entsprechende Rolle hat in Zukunft kein Zugriffsrecht mehr.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um die Einstellungen zu speichern. Mit dem Button **Abbrechen** schließen Sie den Dialog, ohne Änderungen zu speichern.

7.3 Workflow bearbeiten

Um einen vorhandenen Workflow zu bearbeiten, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Workflow** auf den Menüpunkt **Workflow bearbeiten**.

   Es wird die Workflow-Liste geöffnet, in der alle im System eingerichteten Workflows angezeigt werden.

2. Wählen Sie den Workflow, den Sie bearbeiten möchten, durch einen Klick auf den entsprechenden Button **Bearbeiten** aus.

   Es wird das Grid (siehe Seite 90) gestartet und der Workflow dargestellt.

7.4 Workflow löschen

Um einen bestehenden Workflow zu löschen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Workflow** auf den Menüpunkt **Workflow löschen**.

   Es wird die Workflow-Liste geöffnet.

2. Klicken Sie in der Liste auf den Namen des Workflows, der gelöscht werden soll.

Der Workflow wird ohne weitere Sicherheitsabfrage gelöscht und das Menü wird wieder geöffnet.

Das Löschen eines Workflows ist damit abgeschlossen.

7.5 Workflow-PlugIns

Im Folgenden finden Sie eine Liste aller Workflow-PlugIns, die mit **imperia 9** ausgeliefert werden.

Für die meisten PlugIns können Sie einstellen, welche Rollen zum Zugriff berechtigt sind. Lesen Sie hierzu Abschnitt 7.2.4 **Zugriffsrechte für Workflow-Schritte** auf Seite 94. Erläuterungen zu den einzelnen PlugIns und ihrer Konfiguration finden Sie jeweils in einem eigenen Abschnitt.

Grundsätzlich gibt es drei Kategorien von PlugIns:

- interaktive PlugIns, die eine Benutzereingabe erwarten,
- nicht interaktive PlugIns, die keine Benutzereingabe benötigen,
- semi-interaktive PlugIns, die unter manchen Voraussetzungen eine Benutzereingabe benötigen und unter manchen nicht.

Anhand des Icons können Sie erkennen, ob ein Plugin interaktiv, nicht interaktiv oder semi-interaktiv ist. Interaktive Plugins werden durch zwei weiß hinterlegte Figuren ergänzt; semi-interaktive durch eine einzelne und nicht interaktive zeigen Figur in ihrem Icon.

Das Bearbeiten-PlugIn ist ein typisches Beispiel für ein interaktives PlugIn. Es erwartet, dass ein Benutzer das Dokument editiert. Die Grid-Icons dieser PlugIns werden durch zwei weiße Figuren ergänzt.

Ein Beispiel für ein nicht interaktives PlugIn ist das Auto-Expiry-PlugIn, das keinerlei Benutzereingabe erwartet, sondern für jedes Dokument einen Zeitpunkt definiert, an dem es gelöscht werden soll.

Semi-interaktive PlugIns finden sich im Zusammenhang mit der Erstellung von Asset-Dokumenten. Beispielsweise das PlugIn Asset-Aktionen erfordert nur beim Upload eines einzelnen Assets eine Benutzereingabe. Beim Mehrfachupload erfolgt die Festlegung verfügbarer Assets hingegen automatisch.

Dokumentenfortschritt i9+

In **imperia** Templates ist es nun auch möglich den Dokumentenfortschritt anzuzeigen. (siehe

Einzelheiten hierzu entnehmen Sie bitte dem Kapitel **Darstellung Dokumentenfortschritt** i9+ [../program/program.pdf#tmpl_ticktock] im Entwicklerhandbuch.

) Damit an dieser Stelle im Template nur Workflow-PlugiIns angezeigt werden, die auch für den Autor relevant sind, gibt es die Möglichkeit bei jedem Workflow-PlugIn einzustellen ob dieses Plugin im Dokumentenfortschritt im Template angezeigt werden soll. Die Anzeige der Workflowschritte im Schreibtisch ist davon aber nicht betroffen.

7.5.1 Asset-Aktionen

Dieses PlugIn ist nur in Upload-Workflows für Asset-Dateien anwendbar. In diesem Workflow-Schritt legen Sie fest, welche Aktionen für ein Asset verfügbar sein sollen. Es handelt sich um ein semi-interaktives PlugIn, das nur beim Upload einzelner Assets eine Eingabe vom Benutzer erfordert. Beim Mehrfachupload bekommen alle hochgeladenen Assets automatisch alle auswählbaren Aktionen zugewiesen.

Konfiguration

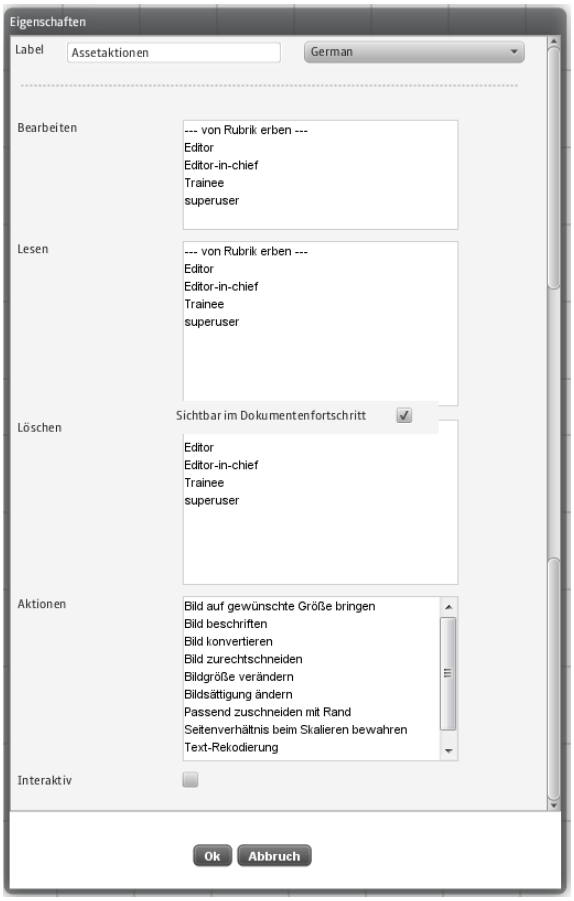


Abb. 7.5: Konfiguration des PlugIns Asset-Aktionen

Folgende Konfigurationsmöglichkeiten stehen zur Verfügung:

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Asset-Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Asset-Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Asset-Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Aktionen

Markieren Sie die Aktionen, die in dem entsprechenden Workflow-Schritt auswählbar sein sollen.

Interaktiv

Wird der Haken gesetzt, kann der Uploader aus den in der Plugin-Konfiguration ausgewählten Aktionen eine erneute Auswahl treffen. Wird der Haken nicht gesetzt, wird dem Uploader keine Auswahl angezeigt und alle hier ausgewählten Aktionen für die Bearbeitung des Assets stehen im MAM zur Verfügung.

Name	Asset-Aktionen
Gruppierung	Interaktiv, Assets,
PlugIn Name	AssetActions.pm

Tabelle 7.1.

7.5.2 Annotate

7.5.2 Annotate

Dieses PlugIn erzeugt aus einem beliebigen Text automatisch einen History-Eintrag. Jedes Dokument, das vom PlugIn bearbeitet wird, erhält den gleichen History-Eintrag. Den Text des History-Eintrags legen Sie in der PlugIn-Konfiguration im Feld **History-Eintrag** fest.

Konfiguration

Im Feld „History Eintrag“ geben Sie den Text ein der an dieser Stell im Workflow in die History geschreiben werden soll.

Name	History-Eintrag
Gruppierung	Information
PlugIn Name	Annotate.pm

Tabelle 7.2.

7.5.3 Archiv-Bereiniger

imperia archiviert bei einem Durchlauf des Workflows den jeweiligen Stand des Dokuments bei jedem Schritt. Es wird also für jeden Workflow-Schritt eine Archiv-Version des Dokuments erzeugt. Je umfangreicher ein Workflow ist, desto mehr archivierte Dokument-Versionen fallen an. Auf diese Weise kann das Archiv schnell sehr umfangreich werden, so dass es viel Speicherplatz belegt, unübersichtlich wird und seine Performance sich mehr und mehr verlangsamt.

Abhilfe kann hier der Archiv-Bereiniger schaffen. Mit diesem PlugIn können Sie bei jedem Durchgang des Workflows das Archiv automatisch von nicht beendeten Versionen des Dokuments bereinigen. Für die Konfiguration stehen Ihnen folgende Optionen zur Verfügung:

Konfiguration

Abb. 7.6: Konfiguration des PlugIns Asset-Aktionen

Aktuelle nicht abgeschlossene Versionen

Wenn Sie diese Option auswählen, werden alle nicht abgeschlossenen Versionen des Dokuments gelöscht, die vor der letzten beendeten Version erstellt wurden. Danach erstellte unbeendete Versionen bleiben erhalten.

Alle nicht abgeschlossenen Versionen

Bei dieser Option werden auch die unbeendeten Archiv-Versionen gelöscht, die nach der letzten beendeten Version erstellt wurden.

Alte nicht abgeschlossene Versionen

Wählen Sie diese Option, um nur die unbeendeten Versionen des Dokuments aus dem Archiv zu löschen, die vor der vorletzten beendeten Version erstellt wurden.

Name	Archiv-Bereiniger
Gruppierung	Aufräumen
PlugIn Name	ArchiveCleaner.pm

Tabelle 7.3.

7.5.4 Publizieren und Löschen

Dieses PlugIn verleiht einem Dokument automatisch das aktuelle Datum als Ablaufdatum. Es wird also nach Beendigung des Workflows sofort gelöscht. Dafür muss auf dem Produktionssystem der Hintergrund-Daemon laufen. Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Publizieren und Löschen
Gruppierung	Steuerung
PlugIn Name	AutoExpiry.pm

Tabelle 7.4.

7.5.5 Publizieren nach Workflow

Dieses PlugIn verleiht einem Dokument automatisch das aktuelle Datum als Freischaltdatum. Es wird also nach Beendigung des Workflows sofort auf das Zielsystem freigeschaltet. Dafür muss auf dem Produktionssystem der Hintergrund-Daemon laufen. Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Publizieren nach Workflow
Gruppierung	Steuerung
PlugIn Name	AutoPublisher.pm

Tabelle 7.5.

7.5.6 Bearbeiten

Dieses PlugIn öffnet das Template zur Eingabe von Inhalten für den Benutzer. Eingegebene Daten können abgespeichert oder verworfen werden.

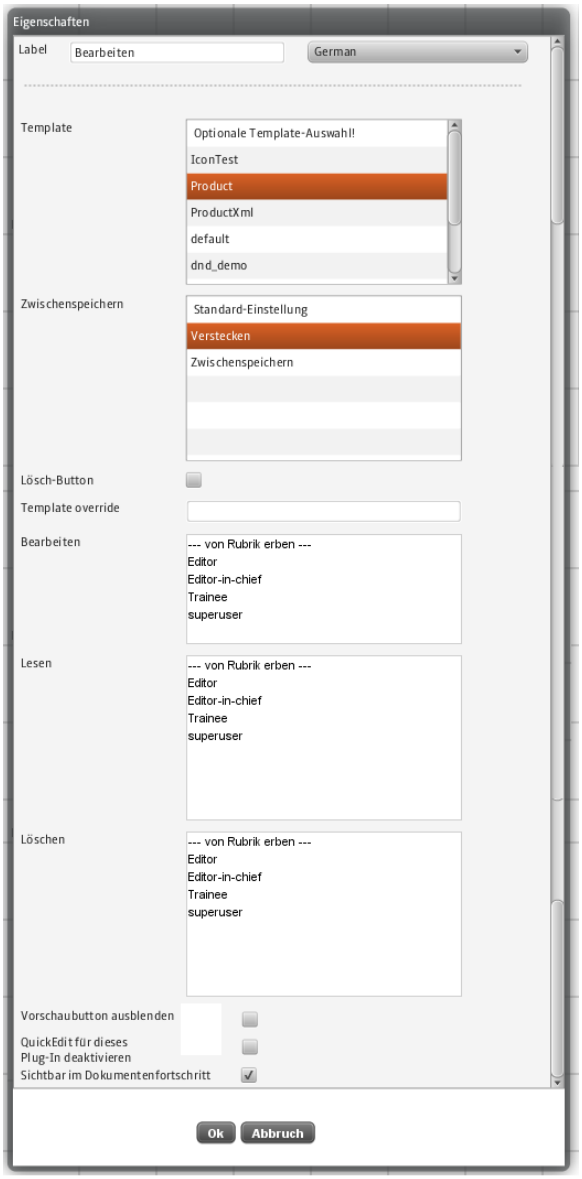


Abb. 7.7: Konfiguration des Bearbeiten-PlugIns

Folgende Konfigurationsmöglichkeiten stehen zur Verfügung:

Template

Über diese Auswahlliste können Sie festlegen, dass für diesen Schritt ein anderes Template verwendet werden soll als in den Rubrikeinstellungen definiert wurde.

Zwischenspeichern

Über diese Auswahlliste können Sie eine Schaltfläche im Template ein- oder ausblenden, welche es erlaubt das Dokument zu speichern ohne zum Schreibtisch zurück zu wechseln.

Lösch-Button

Über diese Schaltfläche können Sie das geöffnete Dokument einschließlich Archiv und zugehörige Objekte löschen.

Template Override

Diese Option dient Ihnen dazu, ein vorausgewähltes Template durch ein anderes zu ersetzen. Sie können hier auch XX-Variablen verwenden, um beispielsweise einen Templatenamen mit einem variablen Suffix zu ergänzen.

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Vorschaubutton ausblenden

Blendet die Schaltfläche zur Vorschau eines Dokumentes im Template aus.

QuickEdit für dieses PlugIn deaktivieren

Ist diese Option gesetzt, wird das betroffene Plugin bei der Bearbeitung mittels der One-Click-Edit Toolbar übersprungen.

Zusätzlich können Sie das Verhalten des Zwischenspeichern konfigurieren. Dabei gibt es drei Möglichkeiten:

- Mit dem Wert `Standard-Einstellung` wird die Einstellung aus der Systemkonfiguration übernommen.
- Mit dem Wert `Verstecken` wird das Zwischenspeichern in diesem Schritt deaktiviert, auch wenn in der Systemkonfiguration etwas andeses steht.
- Mit dem Wert `Zwischenspeichern` wird das Zwischenspeichern in diesem Schritt aktiviert auch wenn in der Systemkonfiguration etwas andeses steht.

Weitere Einzelheiten hierzu entnehmen Sie bitte dem Kapitel **Variablen zur System-Konfiguration** [../admin/admin.pdf#adminConf] im Administrationshandbuch.

Name	Bearbeiten
Gruppierung	Interaktiv, Erfassung
PlugIn Name	Edit.pm

Tabelle 7.6.

7.5.7 Bearbeiten +

Dieses PlugIn ist eine Erweiterung des Bearbeiten-PlugIns. Es bietet einen zusätzlichen Ausgang (Süden), der verwendet wird, wenn ein Dokument gespeichert wird. Die Funktion des Ausgangs im Osten ist identisch mit der Funktion des östlichen Ausgangs des Bearbeiten-PlugIns.

Das „Bearbeiten +“-PlugIn kann beispielsweise verwendet werden, um den Prozess zum Erzeugen von Inhalt statistisch zu erfassen.

Beispiel:

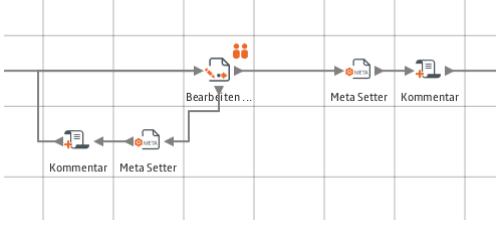


Abb. 7.8: Beispiel Bearbeiten + PlugIn

Dieser Ausschnitt aus einem Workflow bildet folgende Funktion ab:

Jedesmal, wenn ein Dokument bearbeitet und gespeichert wird, verlässt es den Workflow-Schritt durch den südlichen Ausgang und durchläuft dann das MetaSetter-PlugIn und anschließend das Annotate-PlugIn.

Danach wartet es auf die nächste Bearbeitung. Wird der Workflow-Schritt abgeschlossen, verlässt das Dokument den Workflow-Schritt durch den östlichen Ausgang, durchläuft ebenfalls diese beiden PlugIns (damit die gleichen Aktionen wie beim Speichern durchgeführt werden) und gelangt anschließend in den nächsten Workflow-Schritt.

Die Konfiguration entspricht der des Bearbeiten-PlugIns.

Name	Bearbeiten +
Gruppierung	Interaktiv, Erfassung
PlugIn Name	EditOnce.pm

Tabelle 7.7.

7.5.8 Transformation

Mit diesem PlugIn können mit speziellen Transformations-PlugIns anpassungen an den Metainformationen durchgeführt werden.

Konfiguration

Abb. 7.9: Konfiguration des PlugIns Transform

Name	Transformation
Gruppierung	Steuerung
PlugIn Name	Transform.pm

Tabelle 7.8.

7.5.9 Inhaltsprüfung

Mit diesem PlugIn können Sie die Inhalte von Metafeldern überprüfen und bei Abweichungen vom Soll-Zustand einen anderen Workflow-Pfad einschlagen.

Name	Inhaltsprüfung
Gruppierung	Steuerung
PlugIn Name	InputChecker.pm

Tabelle 7.9.

7.5.10 HTML-Konverter

Mit diesem PlugIn können Sie Ihr Dokument mit dem aktuellen Template erstellen und an ein Konvertierungsprogram, zur Umwandlung in ein anderes Datenformat, übergeben.

Konfiguration

Name	HTML-Konverter
Gruppierung	Erfassung
PlugIn Name	HtmlConvert.pm

Tabelle 7.10.

7.5.11 Asset-Variante

Mit diesem PlugIn können Sie automatisch Asset-Varianten beim Hochladen in das MAM erzeugen.

Konfiguration

Abb. 7.10: Konfiguration des PlugIns Asset-Variante

Name	Asset-Variante
Gruppierung	Interaktiv, Assets
PlugIn Name	AssetVariant.pm

Tabelle 7.11.

7.5.12 Kopie erstellen

Das "Kopie erstellen" Workflow-PlugIn erzeugt eine (nahezu) identische Kopie eines **imperia** Dokumentes. Anders als bei **imperia** Copyseiten ist das neue Dokument ein eigenständiges Dokument und somit komplett unabhängig. Veränderungen am Eltern-Dokumentes haben somit keinen Einfluss auf das neu erzeugte Kind-Dokument. Es besteht jedoch eine "lockere" Beziehung zwischen den Dokumenten, welche den Umgang erleichtert. Es besteht die Möglichkeit über die Meta-Variablen *__imperia_parent* bzw. *__imperia_children* die Kind- respektive Eltern-Dokumente auszulesen.

Grundsätzlich werden beim Erstellen des Kind-Dokumentes alle Meta-Variablen des Eltern-Dokumentes übernommen. Diese können jedoch nach Belieben geändert oder auch gelöscht werden. Dabei ist wichtig zu beachten, dass alle durchgeführten Änderungen in genau der Reihenfolge durchgeführt werden, in welcher sie in der PlugIn Konfiguration auftauchen. Zudem werden Änderungen immer auf Basis der aktuell gültigen Meta-Variablen und deren Werten durchgeführt. Wird beispielsweise die Meta-Variable *directory* gelöscht, kann sie später nicht mehr referenziert werden.

Das neu erzeugte Dokument unterscheidet sich in den folgenden Meta-Variablen von dem Elter-Dokument:

- **__imperia_node_id**: Die Meta-Variable *__imperia_node_id* enthält die Node-Id des neu erzeugten Dokumentes und nicht die des Elter-Dokumentes.
- **__imperia_created, __imperia_modified**: Beide Meta-Variablen enhalten die Zeit in Sekunden für den Zeitraum seit dem 1 Januar 1904 GMT 00:00:00 für MacOS bzw 1 Januar 1970 GMT 00:00:00 für andere Betriebssysteme.

- **__imperia_uid**: Enthält die Benutzer-Id des aktuell angemeldeten Benutzers.
- **__imperia_meta_expand**: Ist die Meta-Variable directory vorhanden, enthält die Meta-Variable *__imperia_meta_expand* den Wert 1, welcher durch das MetaEdit-PlugIn gesetzt wurde. Wenn vorhanden und nicht Null wird die Meta-Datei-Direktive *<!--XX-METAMODE-->* true.
- **__imperia_parent**: Diese Meta-Variable enthält die Node-Id des Eltern-Dokumentes.
- **__imperia_children**: Diese Meta-Variable enthält alle Node-Ids der vorhandenen Kind-Dokumente. Beachten Sie, dass diese Variable nur im Eltern-Dokument vorhanden ist.

Zudem stehen die folgenden Konfigurationsmöglichkeiten zur Verfügung:

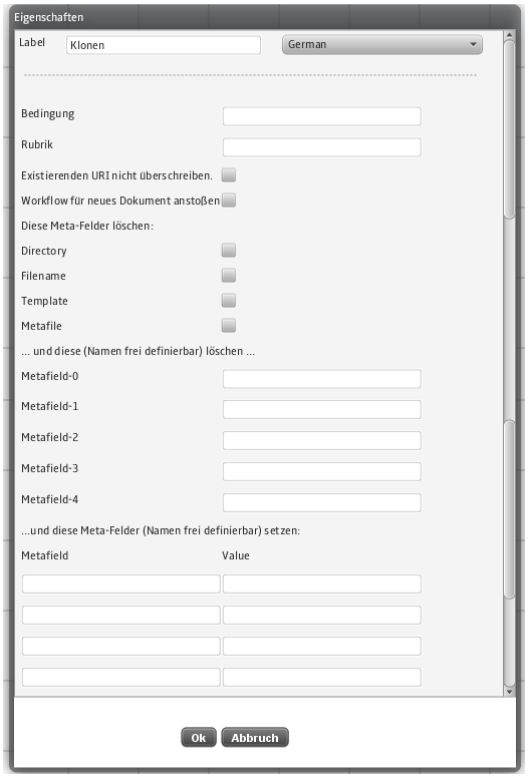


Abb. 7.11: Konfiguration des PlugIns "Kopie erstellen"

Bedingung

Ein Dokument kann niemals ohne eine Bedingung erzeugt werden, um Endlosschleifen im Workflow zu verhindern. Es muss also eine Bedingung angegeben werden, die den Wert einer Variable überprüft um den Clone-Prozess starten zu können. Die Bedingung wird von der **imperia** Template Engine als IF Anweisung interpretiert und muss den gängigen Syntaxregeln genügen.

Beispiel:

Wenn sie die Bedingung

```
"<!--XX-create_copy-->" EQ "yes"
```

eingeben, wird der folgenden Template Code evaluiert:

```
#IF ("<!--XX-create-copy-->" EQ "yes">)
  Dokument wird geclont...
#ELSE
  Dokument wird nicht geclont...
#ENDIF
```

Das PlugIn überprüft nicht den Inhalt der Bedingung. Sind Sie sich sicher, keine Endlosschleife zu erzeugen, können die eine Bedingung angeben, welche immer "true" ergibt (Beispiel: "*1*" oder "*Ich weiss was ich tue*").

Rubrik

Wie jedes **imperia** Dokument muss auch ein geklontes Dokument in einer Rubrik gespeichert werden. Wird im Feld Rubrik nichts angegeben, speichert **imperia** das Dokument in der Rubrik des Eltern-Dokumentes. Andernfalls muss eine Rubriken-Id der Form "*/1/10*" angegeben werden. Anstatt der Rubriken-Id kann auch der Rubrikenname angegeben werde. Dies hat jedoch einen negativen Einfluss auf die Performance, da zunächst nach dem Namen der Rubrik gesucht werden muss, was je nach Größe des Rubrikenbaums lange dauern kann. Zudem kann es zu Problemen kommen, wenn zwei Rubriken mit dem gleichen Namen existieren.

Existierende URI nicht überschreiben

Ist diese Option ausgewählt, versucht **imperia** bereits bestehende Dokumente nicht zu überschreiben.

Workflow für neues Dokument anstoßen

Im Normalfall wird das neue Dokument erzeugt und danach nicht weiter bearbeitet. Ist diese Option jedoch gewählt, wird das Dokument durch den Workflow "geschoben" und stoppt erst vor dem ersten Workflow-PlugIn welches Benutzerinteraktion erfordert. (solche Workflow-PlugIns sind durch zwei kleine Figuren in der Rechten oberen Iconecke gekennzeichnet)

Diese Metafelder löschen ...

Die Meta-Variablen *directory*,*filename*,*template* und *metafile* können mit Informationen, welche aus der Rubrikendefinition ausgelesen werden initialisiert werden. Dies geschieht durch das wählen der jeweiligen Option. Werden diese Optionen nicht gewählt, werden die Werte aus dem Eltern-Dokument übernommen. Wird die Meta-Variable *directory* gelöscht, wird ebenfalls *__imperia_meta_expand* gelöscht (Details zu dieser Variable wurden bereits zuvor erläutert).

... und diese (Namen frei definierbar) löschen

Hier können bis zu zehn frei wählbare Meta-Variablen angegeben werden, welche ebenfalls aus dem neuen Dokument gelöscht werden sollen. Hinweis: An dieser Stelle können **imperia** Template Variablen verwendet werden.

... und diese Meta-Felder (Namen frei definierbar) setzen:

Hier können bis zu zehn frei wählbare Meta-Variablen und dessen Werte angelegt werden. Existieren diese Variablen bereits werden sie überschrieben. Hinweis: An dieser Stelle können **imperia** Template Variablen verwendet werden.

Name	Kopie erstellen
Gruppierung	Steuerung
PlugIn Name	Clone.pm

Tabelle 7.12.

7.5.13 Dokumentenauswahl

Das Dokumentenauswahl-PlugIn hilft Ihnen bei der Behandlung von URI-Konflikten eines **imperia** -Dokuments.

Sie können mit **imperia** eine beliebige Anzahl von Dokumenten mit gleichem URI anlegen. Allerdings kann sich der URI eines Dokumentes ändern, wann immer es den Workflow durchläuft.

Wenn Sie mehrere Dokumente mit gleichem URI anlegen, ergibt sich daraus, dass im Document-Root eine Datei überschrieben wird. Oft ist dieser Effekt erwünscht. Manchmal ergibt sich daraus jedoch eine Beeinträchtigung für den Administrator, der den tatsächlichen Inhalt der Datei überwachen muss.

Mit dem Dokumentenauswahl-PlugIn können Sie die Behandlung von URI-Konflikten an die spezifischen Anforderungen Ihrer Site anpassen. Dies ist auf Rubrikenebene möglich.

Jedesmal, wenn ein Dokument den Workflow-Schritt mit dem PlugIn durchläuft, wird dessen Rubrik nach Dokumenten mit dem selben URI durchsucht. Dabei können folgende Fälle eintreten:

- Es ist kein Dokument mit dem selben URI vorhanden. Nichts passiert, der Workflow läuft ganz normal weiter.
- Es existieren ein oder mehrere Dokumente mit dem selben URI. Sie können den Konflikt dann interaktiv auflösen. Für den Vergleich der URIs werden symbolische Links und Anomalien wie doppelte Slashes im Pfad normalisiert.

Für die Auflösung von URI-Konflikten stehen im Dialogfenster folgende Möglichkeiten zur Auswahl:

- **Vorschau**: Durch Klick auf den verlinkten Dokumententitel können Sie sich eine Vorschau des Dokumentes in einem neuen Browserfenster anzeigen lassen.
- **Status**: Ein Dokument befindet sich entweder "im Workflow" oder "im Archiv".
- **Bearbeiten**: Sie verwerfen das aktuelle Dokument und wählen stattdessen ein bestehendes zur Bearbeitung aus. Dabei wird das aktuelle Dokument komplett gelöscht und das ausgewählte durchläuft den Workflow. Sie führen dann den entsprechenden Workflow-Schritt des jeweiligen Dokuments aus. Bei Dokumenten, die aus dem Archiv importiert werden, ist das der auf den DocSelector-Schritt folgende Workflowschritt. Dokumente, die sich bereits im Workflow befinden, bearbeiten Sie an der entsprechenden Stelle weiter.
- **Kopieren**: Sie kopieren den Inhalt des ausgewählten Dokuments in das aktuelle und gehen weiter zum nächsten Workflow-Schritt. Wählen Sie diese Möglichkeit, um ein neues Dokument zu erstellen, dessen Inhalt auf dem einer älteren Version basiert.
- **Neues Dokument**: Sie verlassen den Dokumentenauswahl-Schritt ohne weitere Aktion. Das aktuelle Dokument gelangt ohne Änderung zum nächsten Workflow-Schritt.

Die Konfiguration des PlugIns nehmen Sie im Workflow-Grid vor. Dafür stehen Ihnen folgende Parameter zur Verfügung:

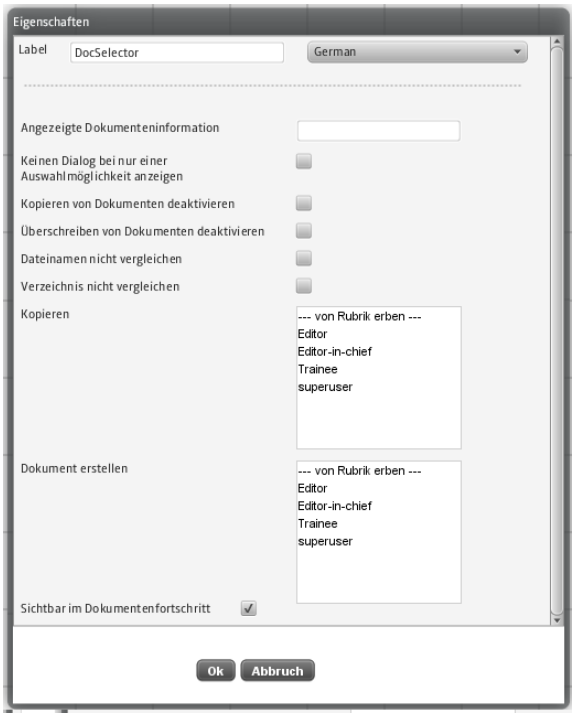


Abb. 7.12: Konfiguration des PlugIns DocSelector

Angezeigte Dokumenteninformationen:

Normalerweise wird in der Liste existierender Dokumente jeweils der Titel angezeigt bzw. "**[ohne Titel]**", wenn ein Dokument keinen Titel hat. Da alle Workflow-Parameter der Template-Expandierung unterworfen sind, entspricht die Anzeige des Titels einem Zugriff auf *<!--XX-title-->*. Wenn Sie beispielsweise den Eigentümer des Dokumentes hinter dem Titel in Klammern anzeigen lassen möchten, ändern Sie dies wie folgt um:

```
<!--XX-title--> (Benutzer: <!--USER_CONF:name-->)
```

Dialog bei einfacher Auswahl nicht anzeigen:

Mit dieser Option aktivieren Sie folgendes Verhalten:

1. Existiert bereits genau ein Dokument mit identischem URI und kann das Dokument nicht kopiert werden, wird die letzte Version des Dokuments in den Workflow importiert. Dabei wird kein interaktiver Dialog angezeigt und das aktuelle Dokument wird automatisch komplett gelöscht.

   Dieser Fall tritt entweder ein, wenn Sie keine Berechtigung zur Erstellung von Kopien in diesem Workflow-Schritt haben oder die Erstellung von Kopien allgemein deaktiviert wurde.

   Tatsächlich unterbinden Sie so die Entstehung von mehreren Dokumenten mit identischem URI in der betreffenden Rubrik. Wenn Sie ein neues Dokument erstellen, importieren Sie stattdessen automatisch die letzte Version aus dem Archiv und bearbeiten diese, wobei das neue Dokument automatisch gelöscht wird.

   Hinweis:

   Diese Variante eignet sich besonders, wenn Sie nur ein Dokument in einer Rubrik anlegen und dieses im Laufe der Zeit ändern möchten. Da immer die letzte Version automatisch aus dem Archiv importiert und überschrieben wird, sorgen Sie auch dafür, dass nur wenige archivierte Versionen des Dokumentes vorliegen.

   Zugriffe auf die Menüpunkte "Schreibtisch" oder das Archiv sind dabei nicht nötig. Auf diese Weise minimieren Sie den Schulungsaufwand für Ihre Autoren und beschleunigen gleichzeitig die Arbeitsabläufe, da beispielsweise Zugriffe auf das Archiv je nach Umfang des Projektes sehr ressourcenintensiv sein können.

   Wenn Sie garantieren möchten, dass in einer Rubrik nicht mehrere Dokumente den selben URI haben, wählen Sie außerdem die Optionen "*Kopieren von Dokumenten deaktivieren*" (siehe Abschnitt 7.5.13, „Kopieren von Dokumenten deaktivieren:“ und Abschnitt 7.5.13, „Überschreiben von Dokumenten deaktivieren:“). Gegenüber einer Einschränkung der Rollenberechtigungen hat dies den Vorteil, dass Benutzer mit Zugriff auf die Superuser-Rolle diese Regel nicht unbeabsichtigt umgehen können.

2. Sollte unter der URI nicht genau ein Dokument existieren, bekommen Sie den Dialog zur interaktiven Auflösung des URI-Konfliktes angezeigt.

Kopieren von Dokumenten deaktivieren:

Wenn Sie diese Option markieren, können Sie keine Kopie mit identischem URI auf der Basis eines bestehenden Dokumentes erzeugen. Sie setzen damit alle vorgegebenen Kopierberechtigungen für diesen Workflow-Schritt außer Kraft.

Überschreiben von Dokumenten deaktivieren:

Wenn Sie diese Option markieren, können Sie bereits bestehende Dokumente mit identischem URI bearbeiten oder mit einem neuen Dokument überschreiben. Dabei basiert der Inhalt des neuen Dokumentes auf dem des bereits bestehenden. Mit dieser Option setzen Sie alle etwaigen Berechtigungen zur Erzeugung neuer Dokumente außer Kraft. Diese Einstellung kann auch vom Superuser nicht umgangen werden.

Kopieren:

Wählen Sie hier die Rollen aus, die Kopien auf der Basis bestehender Dokumente erzeugen dürfen.

Wichtig:

Beachten Sie, dass Benutzer mit Zugriff auf die Superuser-Rolle keinerlei Beschränkungen durch diese Option unterliegen.

Dokument erstellen:

Mit dieser Option schränken Sie die Rollen ein, die "**neue Dokumente**" erzeugen dürfen.

Hinweis:

Tatsächlich ist ja bereits ein neues Dokument erzeugt worden, wenn Sie diesen Workflowschritt erreichen. "Dokument erstellen" bedeutet hier also genau genommen, dass das aktuelle Dokument nicht gelöscht wird.

Wichtig:

Beachten Sie, dass Benutzer mit Zugriff auf die Superuser-Rolle keinerlei Beschränkungen durch diese Option unterliegen.

Name	Dokumentenauswahl
Gruppierung	Interaktiv, Steuerung
PlugIn Name	DocSelector.pm

Tabelle 7.13.

7.5.14 Depublizierung

Setzen Sie das Eraser-PlugIn ein, um ein Dokument einschließlich aller eventuell bestehenden Kopien vom Zielsystem zu löschen. Das Dokument kann anschließend weiter im Workflow bearbeitet werden. Die Archiv-Version(en) des gelöschten Dokumentes bleiben erhalten. Das PlugIn benötigt keine weitere Konfiguration.

Name	Depublizierung
Gruppierung	Aufräumen
PlugIn Name	Eraser.pm

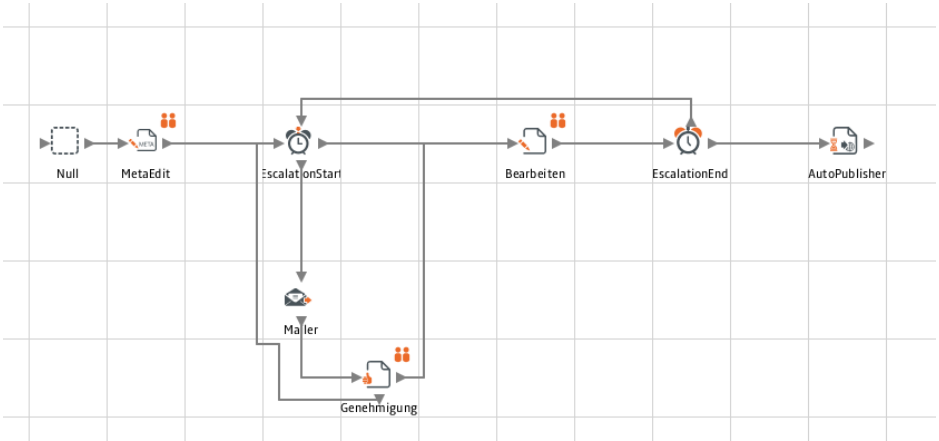
Tabelle 7.14.

7.5.15 Eskalation-Start

Die Aufgabe der Eskalations-PlugIns ist es, zeitliche Vorgaben für das Durchlaufen von bestimmten Workflowschritten zu überprüfen und entsprechend zu reagieren.

Beispiel: Eine Pressemitteilung die in mehreren Schritten bearbeitet wird, muss spätestens 4h nach dem Start der Erstellung den Workflow verlassen und publiziert werden. Wird diese Zeit nicht eingeahlten, so soll der Ersteller der Pressemitteiling eine Mail bekommen um darauf reagieren zu können. In diesem Fall würde man um die Bearbeitungskette eine Eskalation aufbauen in dem man an den Anfang ein Eskalation-Anfang PlugIn einfügt und vor dem Ende ein Eskalation-Ende PlugIn einfügt. Damit die beiden Plugins mit einaner arbeiten können, werden die Nordausgänge verbunden.

Beispiel-Workflow für eine Eskalation.



In diesem Plg-In wird die Zeit eingestellt, nach der „eskaliert“ werden soll. Das Format ist HH::MM::SS (z.B. 01:01:01 -> 1Std, 1min, 1sec)

Eskaliert wird, wenn sich das Dokument zwischen dem Start und Ende Plug-in im Workflow befindet und der definierte Zeitraum seit dem durchlaufen des Start Plug-ins abgelaufen ist. Das Dokument wird dann in den Schritt verschoben der mit Südausgang des Eskalation-Anfang verbunden ist.

Wie schon oben beschrieben verknüpft man zusammengehörige Eskalationen in dem man den Nord Eingang des Startes mit dem Nordausgang des Endes verbidet. Damit lassen sich auch geschachtelte Eskalation realisieren.

Wird bei einer geschachtelten Eskalation die äußere Eskalation getriggert, so wird die innere Eskalation (falls gestartet) annuliert.

Voraussetzung dafür dass der Eskalations-Mechanismus funktioniert ist, dass der Hermes läuft.

Name	Escalation-Start
Gruppierung	Qualitätssicherung
PlugIn Name	EscalationStart.pm

Tabelle 7.15.

7.5.16 Eskalation-Ende

Die PlugIns Eskalation-Start und Eskalation-Ende werden immer paarweise verwendet. Bitte lesen Sie die Funktionsweise in Abschnitt 7.5.15 **Eskalation-Start** auf Seite 108 nach.

Name	Eskalation-Ende
Gruppierung	Qualitätssicherung
PlugIn Name	EscalationEnd.pm

Tabelle 7.16.

7.5.17 Existenzprüfung

Mit dem Existenzprüfung-PlugIn prüfen Sie, ob ein Dokument mit identischem URI bereits den Workflow durchlaufen hat. Das bedeutet, dass es ein anderes Dokument mit den selben Werten in den Metafeldern *directory* und *filename* gibt, dessen Status **beendet** ist.

Trifft diese Bedingung zu, wird das PlugIn durch den Südausgang ("Ja") verlassen, andernfalls durch den Ostausgang ("Nein").

Hinweis:

Beachten Sie, dass es sich dabei um ein **anderes** *Dokument handeln muss. Ist das Dokument mit dem selben URI lediglich eine frühere Version des aktuell im Workflow befindlichen, wird dies so behandelt, als ob es noch kein Dokument mit diesem URI gäbe. Das ist notwendig, um Schwierigkeiten bei der Bearbeitung bereits freigeschalteter Dokumente zu vermeiden.*

Dieses PlugIn benötigt keine Konfiguration.

Ältere **imperia** -Versionen lösen etwaige URI-Konflikte beim Freischaltvorgang einfach durch Überschreiben des bestehenden Dokuments. Mit Hilfe des Exists-PlugIns haben Sie die Möglichkeit, Vorkehrungen für den Fall zu treffen, dass ein Dokument mit dem selben URI existiert. Typischerweise positionieren Sie das PlugIn dort im Workflow, wo der URI des Dokuments sich nicht mehr ändern wird, also "stabil" ist.

Name	Existenzprüfung
Gruppierung	Steuerung
PlugIn Name	Exists.pm

Tabelle 7.17.

7.5.18 Depublikationsdatum

Mit Hilfe dieses PlugIns wird das Ablaufdatum eines Dokuments bestimmt. Der Benutzer erhält beim Erreichen dieses Workflow-Schrittes eine Eingabemaske, in der er das Ablaufdatum über eine Pull-Down-Liste bestimmt. Damit das Dokument beim Erreichen des eingestellten Datums gelöscht wird, muss der Hintergrund-Daemon auf dem Produktions- und Zielsystem laufen.

Konfiguration

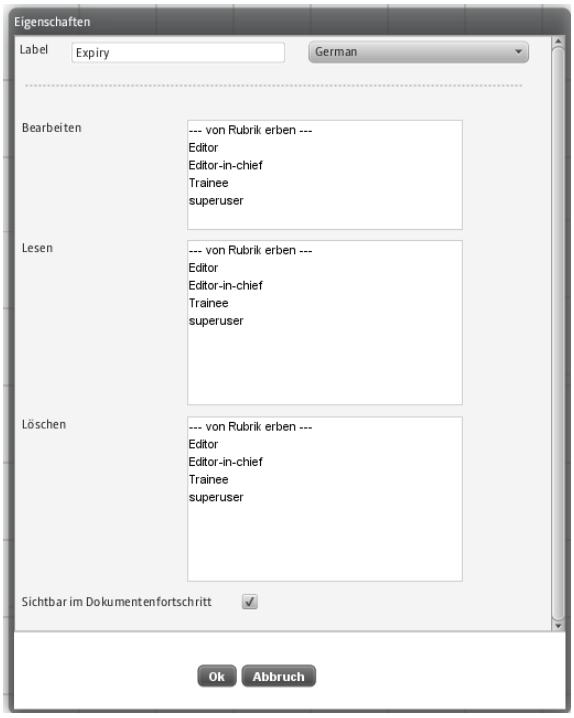


Abb. 7.13: Konfiguration des PlugIns Depublikationsdatum

Im Grid können Sie die Zugriffsberechtigungen für das PlugIn einstellen. Legen Sie fest, welche Rollen ein Dokument in diesem Schritt bearbeiten, lesen oder löschen dürfen.

Name	Depublikationsdatum
Gruppierung	Steuerung, Erfassung
PlugIn Name	Expiry.pm

Tabelle 7.18.

7.5.19 Genehmigung

Dieses PlugIn stellt eine Prüfinstanz für Dokumente dar (Mehr-Augen-Prinzip). Beide Ausgänge des Genehmigung-PlugIns müssen mit einem anderen PlugIn verbunden werden, so dass Dokumente, abhängig davon, ob der Benutzer genehmigt oder verwirft, einen anderen Weg durch den Workflow nehmen.

Wird der südliche Ausgang nicht verbunden, gibt das PlugIn eine Warnung aus, da in einem solchen Fall ein abgelehntes Dokument in der Freischaltliste erscheinen kann. Der südliche Ausgang wird verwendet, wenn eine Genehmigung nicht erteilt wird, der östliche Ausgang wird verwendet, wenn die Genehmigung erteilt wird.

Das PlugIn interpretiert nicht, ob das mit dem östlichen Ausgang verbundene PlugIn nach einer Genehmigung sinnvoll ist. Es bietet eher die Möglichkeit, den Workflow interaktiv zu verzweigen.

Konfiguration

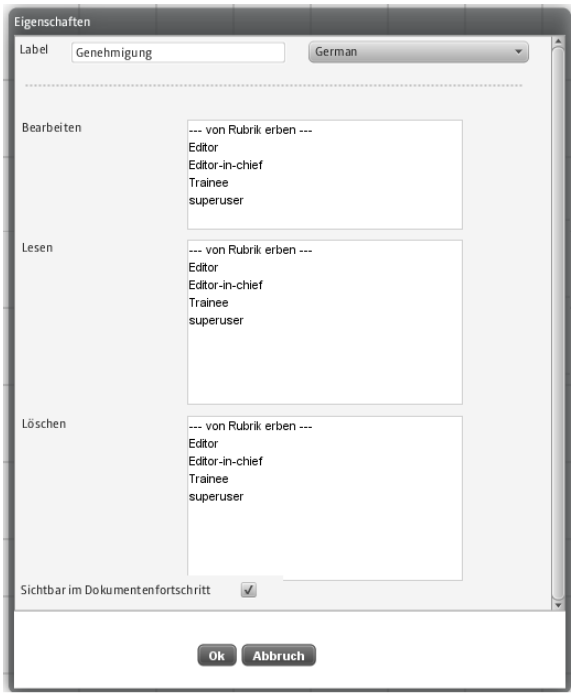


Abb. 7.14: Konfiguration des Genehmigen PlugIns

Das Genehmigung-PlugIn kann im Workflow nicht umgangen werden. Über die Konfiguration des PlugIns steuern Sie, wer den Genehmigungsschritt ausführen darf:

Bearbeiten

Rollen mit dieser Berechtigung dürfen ein Dokument genehmigen bzw. ablehnen.

Lesen

Wählen Sie aus, welche Rollen Dokumente, die sich in diesem Schritt befinden, auf dem Schreibtisch sehen können. Allerdings können nur Benutzer mit dem Bearbeiten-Recht den Workflowschritt auch ausführen.

Löschen

Diese Berechtigung ermöglicht das Löschen von zu genehmigenden Dokumenten vom Schreibtisch.

Lesen Sie hierzu das Kapitel **Dokumente genehmigen** im Benutzerhandbuch.

Name	Genehmigung
Gruppierung	Interaktiv, Qualitätssicherung
PlugIn Name	Approval.pm

Tabelle 7.19.

7.5.20 Bedingung

Dieses PlugIn wird benutzt, um den Weg von Dokumenten in Abhängigkeit von einer oder mehreren Bedingungen zu bestimmen. Es besitzt drei Ausgänge und benötigt keine Benutzereingabe.

Trifft die erste Bedingung zu, wird der nördliche Ausgang des PlugIns verwendet. Trifft die erste Bedingung nicht zu, wird die zweite Bedingung überprüft.

Trifft die zweite Bedingung zu, wird der östliche Ausgang benutzt. Trifft die zweite Bedingung ebenfalls nicht zu, wird der südliche Ausgang benutzt.

Abb. 7.15: Eigenschaften des Bedingung-PlugIns

Die Bedingungen werden im Eigenschaften-Dialog des PlugIns eingegeben. Die Syntax hierfür ist weitgehend identisch mit der Syntax für IF-Abfragen im Template. Sie müssen lediglich die Klammern weglassen.

Bleibt eine der Eingabezeilen leer, gilt die damit verbundene Bedingung als nicht erfüllt.

Achten Sie bei der Eingabe der Bedingungen auf Leerzeichen, Tabstopps, Zeilenumbrüche und ähnliches. Leerzeichen am Anfang und am Ende einer Eingabezeile werden nicht beachtet. Mehrere Leerzeichen innerhalb des Bedingungstextes werden zu einem Leerzeichen zusammengefasst.

Name	Bedingung
Gruppierung	Steuerung
PlugIn Name	IfElse.pm

Tabelle 7.20.

7.5.21 Genehmigung +

Dieses PlugIn stellt eine Kombination der Link-Check-Funktion, die über den Schreibtisch verfügbar ist, und dem Genehmigungs-PlugIn dar.

Konfiguration

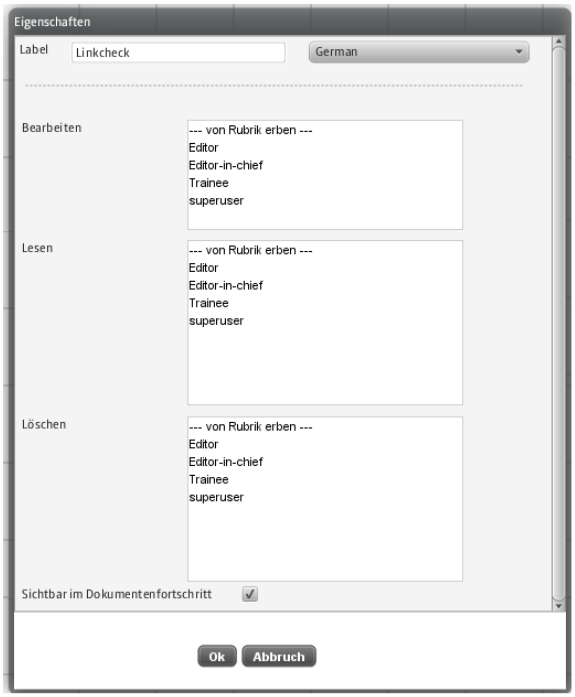


Abb. 7.16: Konfiguration des PlugIns Genehmigung +

Führen Sie den Workflow-Schritt aus, um den Dialog **Link-Checker** zu öffnen:

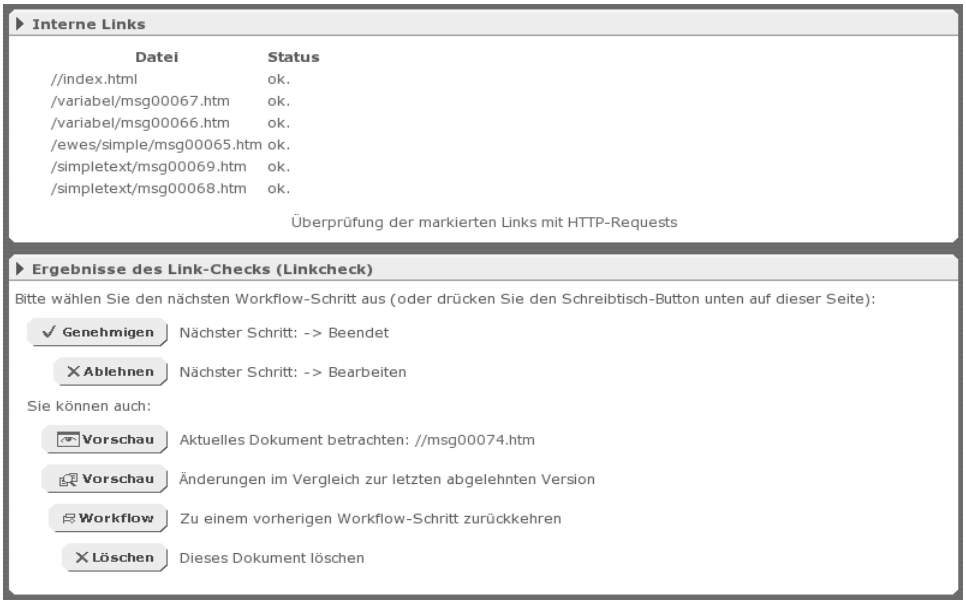


Abb. 7.17: Dialog Inhaltsprüfung

Interne Links

Dieser Bereich zeigt das Ergebnis der Linküberprüfung. Die aufgeführten Dateien sind verlinkt, so dass Sie diese im Browser aufrufen können. Dateien, bei denen die Linkprüfung fehlgeschlagen ist, können Sie mit einer nebenstehenden Checkbox markieren und anschließend für die markierten Dateien eine **Überprüfung der markierten Links mit HTTP-Requests** anstoßen. Klicken Sie hierzu auf den entsprechend benannten Link in diesem Bereich.

Ergebnisse des Link-Checks (Linkcheck)

Die hier angezeigten Bedienelemente entsprechen denen des Genehmigungs-PlugIns. Lesen Sie hierzu auch das Kapitel **Dokumente genehmigen** im Benutzerhandbuch.

Genehmigte Dokumente verlassen den Workflow-Schritt über den Ostausgang und gelangen zum nachfolgenden Schritt. Abgelehnte Dokumente verlassen diesen Schritt über den Südausgang.

Alternativ können Sie auch auf den Button **Schreibtisch** klicken, um zum Schreibtisch zurückzugelangen. Das Dokument bleibt dann im Link-Check. Eine Weiterleitung zum nächsten Workflow-Schritt ohne die Genehmigung des Dokuments ist nicht möglich.

Konfiguration des PlugIns

In der Konfiguration können Sie festlegen, welche Rollen zur Ausführung dieses Schritts berechtigt sind.

Name	Genehmigung +
Gruppierung	Interaktiv, Qualitätssicherung
PlugIn Name	LinkChecker.pm

Tabelle 7.21.

7.5.22 Mailversand

Dieses PlugIn erlaubt den einfachen Versandt von E-Mails.

Wichtig:

Um dieses Plugin betreiben zu können ist es notwendig das Perlmodul HTML::FormatText::WithLinks und Email::Address zu installieren!

Hinweis:

Es kann erforderlich sein den Java- und Browsercache zu leeren, sollte es zu Fehlern in der Darstellung neu installierter Workflowplugins kommen.

Hinweis:

Dieses Plugin muss extra erworben werden und ist nicht im normalen Installationpaket enthalten!

Hinweis:

Um dieses Plugin verwendet zu können muss die system.conf Variable `SMTP_SERVER` *mit dem zu verwendenden SMTP-Server gesetz sein!*

Konfiguration

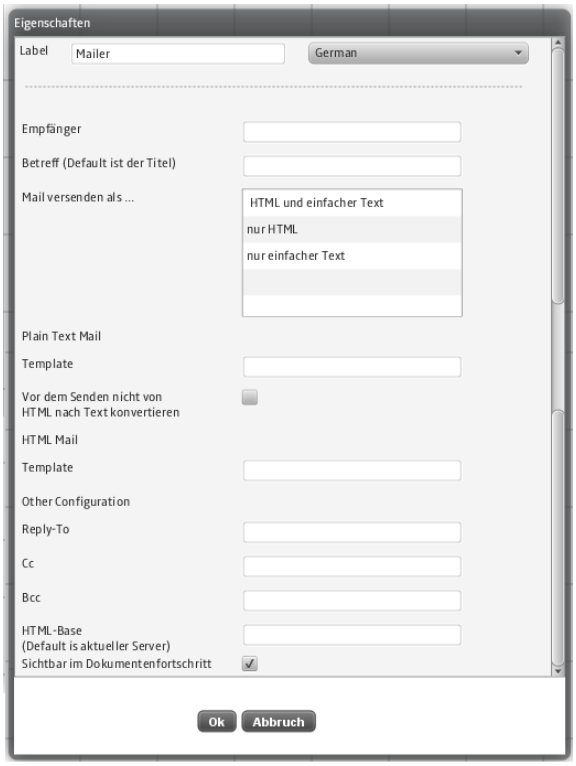


Abb. 7.18: Eigenschaften des Mailversand-PlugIns

Folgende Konfigurationsmöglichkeiten stehen zur Auswahl:

Absender

In diesem Feld kann der Absender der versendeten E-Mail angegeben werden. Ist das Feld nicht gesetzt wird die E-Mailadresse des aktuellen Benutzers verwendet. Sollte keine der beiden Varianten existieren, wird die Fallback E-Mailadresse des Webservers verwendet.

In diesem Feld kann **imperia** Templatecode verwendet werden.

Empfänger

Hier muss der Empfänger der E-Mail angegeben werden. Mehrere Empfänger können durch Komma getrennt werden. Dabei sind die folgenden Formate erlaubt:

- `john.doe@imperia-ag.de`
- `<john.doe@imperia-ag.de>`
- `John Doe <john.doe@imperia-ag.de>`
- `"John Doe" <joh.doe@imperia-ag.de>`

Empfänger die in diesem Feld angegeben werden, sind für alle andere Empfänger sichtbar.

In diesem Feld kann **imperia** Templatecode verwendet werden.

Betreff

Hier kann der Betreff der E-Mail angegeben werden. Ist das Feld nicht gesetzt, wird der der Dokumententitel verwendet.

Mail versenden als ..

Sollte in diesem Feld keine Angabe gemacht werden, wird die E-Mail als HTML- und Plaintextmail verschickt, andernfalls nur die gewählte Option.

Template

In den Templatefeldern können die Namen von Templates angegeben werden, welche zum Erstellen der Mail verwendet werden sollen. Wir kein Template angegeben wird das Template des Dokumentes verwendet. Wurde das Template verarbeitet, wird automatisch das Metafeld `__imperia_mailer` auf `1` gesetzt.

Vor dem Senden nicht von HTML nach Text konvertieren

Sollte diese Option nicht gewählt werden, nimmt **imperia** an, dass die Ausgabe des Templateprozessores HTML ist. Daraus folgt, dass die weitere Ausgabe im Plaintext Format erfolgt. Das Setzen dieser Option macht nur dann Sinn, wenn das Template selber Plaintext erzeugt.

Antwort an

Die Antwortadresse in der E-Mail.

In diesem Feld kann **imperia** Templatecode verwendet werden.

Cc

Ein weiterer Empfänger der E-Mail. Empfänger die in diesem Feld angegeben werden, sind für alle anderen Empfänger sichtbar.

In diesem Feld kann **imperia** Templatecode verwendet werden.

Bcc

Ein weiterer Empfänger der E-Mail. Empfänger die in diesem Feld angegeben werden, sind NICHT für andere Empfänger sichtbar.

In diesem Feld kann **imperia** Templatecode verwendet werden.

HTML Base

imperia wird versuchen das `<base>` tag in der HTML Version der E-Mail zu setzen. Der Inhalt des Tags ist die Adresse des Webservers. In diesem Feld kann eine andere Serveradresse der Form `http://www.imperia-ag.de` angegeben werden.

Sichtbar im Dokumentenvortschritt

Ist diese Option gewählt, wird das Plugin im Dokumentenvortschritt angezeigt.

Mailserver und Parameter wird in der SYSTEM-Conf eingetragen

Name	Mailversand
Gruppierung	Information
PlugIn Name	Mailer.pm

Tabelle 7.22.

7.5.23 Metaedit

Dieses PlugIn zeigt die Eingabemaske der Metadatei an und speichert die dort eingegebenen Daten.

Konfiguration

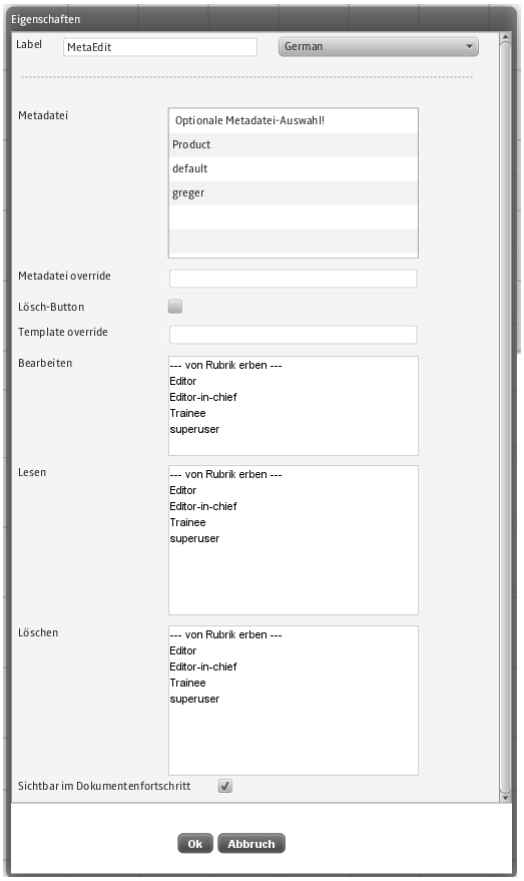


Abb. 7.19: Eigenschaften des Metaedit-PlugIns

Folgende Konfigurationsmöglichkeiten stehen zur Auswahl:

Metadatei

Über eine Auswahlliste können Sie für diesen Workflow-Schritt eine Metadatei auswählen, die von den allgemeinen Einstellungen für die Rubrik abweicht.

Metadatei override

Tragen Sie hier den Namen einer alternativen Metadatei ein, die in diesem Schritt zum Einsatz kommen soll. Sie können hier auch über XX-Variablen Ergänzungen vornehmen.

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Name	MetaEdit
Gruppierung	Interaktiv, Erfassung
PlugIn Name	MetaEdit.pm

Tabelle 7.23.

7.5.24 Metadaten setzten

Dieses PlugIn erlaubt es Ihnen, innerhalb des Workflows beliebigen Metafeldern Werte zu zuweisen. Sowohl der Name als auch der Wert des Metafeldes werden wie in einem Template expandiert.

Beispiel:

```
last_change     <!--Xstrftime:en_US:%c-->
```

Hiermit wird der Wert der Meta-Variablen `last_change` auf Datum und Zeit nach der en_US_Locale gesetzt. Würde beispielsweise eine Meta-Variable `timestamp` o.ä. existieren, könnte man auch als Schlüssel `<!--XX-timestamp-->` verwenden.

Abb. 7.20: Eigenschaften "Metadaten setzten"

Zurzeit können in einem "Metadaten setzten"-PlugIn gleichzeitig zehn Meta-Variablen gesetzt werden. Es ist jedoch nicht möglich, einer einzelnen Meta-Variablen mehrere Werte zu zuweisen.

Name	Metadaten setzten
Gruppierung	Steuerung
PlugIn Name	MetaSetter.pm

Tabelle 7.24.

7.5.25 Metadatenselektion

Das Metadatenselektion-PlugIn bildet die Funktionalität der Perl-Funktion *splice()* zur Bearbeitung von Metadaten ab. Die Konfiguration des PlugIns verhält sich analog zum Aufruf der Perlfunktion. Wie beim Perl-Splice können Sie mit diesem PlugIn je nach Konfiguration also die Funktionen *shift()*, *unshift()*, *pop()* und *push()* auf Daten in Metafeldern anwenden.

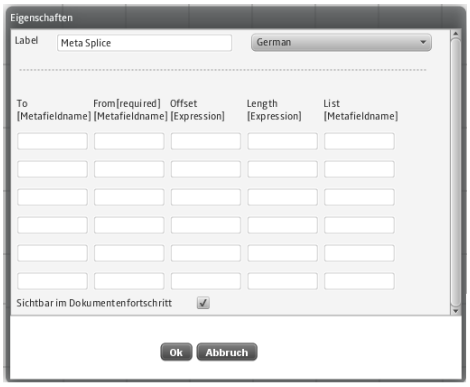


Abb. 7.21: Eigenschaften des PlugIns Metadatenselektion

Damit lassen sich zum Beispiel Workflows mit Schleifen umsetzen, wie das folgende Beispiel zeigt:

Name	Metadatenselektion
Gruppierung	Steuerung
PlugIn Name	MetaSplice.pm

Tabelle 7.25.

7.5.26 Mülleimer

Dieses PlugIn entfernt Dokumente aus dem Workflow ohne weitere Abfrage. Es entfernt jedoch keine archivierten Dokument-Versionen oder zu einem Dokument gehörende Dateien (Bilder, Multimedia-Objekte etc.). Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Mülleimer
Gruppierung	Aufräumen
PlugIn Name	TrashCan.pm

Tabelle 7.26.

7.5.27 Mehrsprachigkeit

Mit dem PlugIn Mehrsprachigkeit steuern Sie die Ersetzungen, die der Templateprozessor für die Verwaltung mehrsprachiger Inhalte vornimmt. Setzen Sie es im Workflow vor dem Edit-PlugIn ein. Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Mehrsprachigkeit
Gruppierung	Steuerung
PlugIn Name	Multilang.pm

Tabelle 7.27.

7.5.28 Platzhalter

Bei diesem PlugIn handelt es sich um eine Beispiel-PlugIn-Implementierung ohne weitere Funktionalität. Es dient als Rumpf für die Entwicklung eigener PlugIns. Ferner kann es dazu verwendet werden, die Übersichtlichkeit eines stark verzweigten Workflows zu verbessern.

Das Platzhalter-PlugIn eignet sich auch als „Hilfs-Workflow-Eingang“, der Änderungen am Anfang des eigentlichen Workflows ermöglicht.

Zu jeder Rubrik, in der Dokumente erstellt werden können, gehört ein Workflow, den die Dokumente der Rubrik durchlaufen. In den Rubrikeinstellungen wird dabei auch immer der Eingang des Workflows gespeichert, beispielsweise „MetaEdit“. Wenn Sie jedoch einen Workflow verändern, indem Sie dem ursprünglichen Eingang ein anderes PlugIn voranstellen, beispielsweise den Dokumentenauswahl, kann die Referenz auf den ursprünglichen Workflow-Eingang nicht mehr aufgelöst werden. Dadurch produzieren alle Rubriken, die den veränderten Workflow nutzen, plötzlich eine Fehlermeldung und es lassen sich darin keine Dokumente mehr anlegen.

Stellen Sie hingegen Ihrem Workflow ein Platzhalter-PlugIn voran, ist es vollkommen unproblematisch, zwischen Platzhalter-PlugIn und eigentlichem Workflow-Beginn weitere Workflow-Schritte einzuhängen. In den Rubriken, die den betreffenden Workflow nutzen, ist das Platzhalter-PlugIn als Eingang gespeichert worden und dies ist auch nach der Änderung der erste Schritt.

Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Platzhalter
Gruppierung	Information
PlugIn Name	Null.pm

Tabelle 7.28.

7.5.29 Publikationssdatum

Dieses PlugIn gibt dem Benutzer die Möglichkeit, ein Veröffentlichungsdatum einzugeben. Das Dokument wird automatisch freigeschaltet, wenn dieses Datum erreicht wird. Dieses PlugIn benötigt einen gestarteten Hintergrund-Daemon auf dem Produktionssystem.

Konfiguration

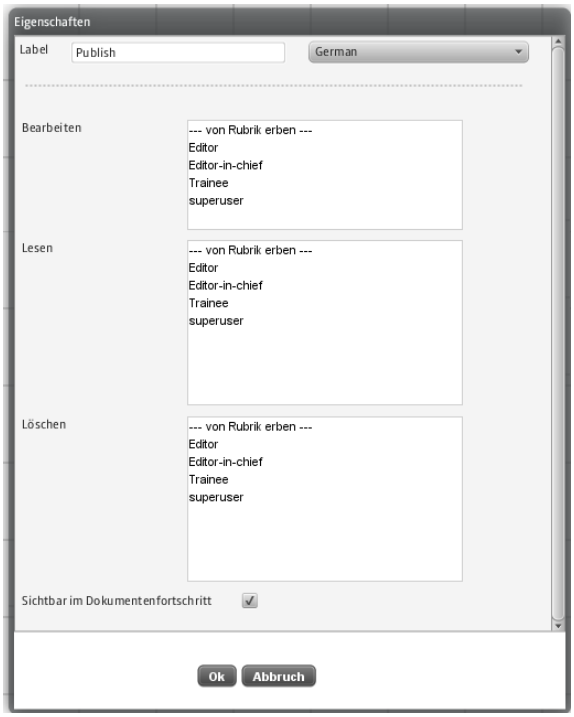


Abb. 7.22: Konfiguration des PlugIns Publikationssdatum

Im Workflow-Grid können Sie die Zugriffsberechtigungen für das PlugIn einstellen. Legen Sie fest, welche Rollen ein Dokument in diesem Schritt bearbeiten, lesen oder löschen dürfen.

Name	Publikationssdatum
Gruppierung	Interaktiv, Steuerung, Erfassung
PlugIn Name	Publish.pm

Tabelle 7.29.

7.5.30 Schredder

Dieses PlugIn löscht Dokumente aus dem Workflow ohne weitere Sicherheitsabfrage. Gleichzeitig werden auch alle archivierten Versionen dieses Dokuments und alle eingebetteten Objekte entfernt.Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Schredder
Gruppierung	Aufräumen
PlugIn Name	Shredder.pm

Tabelle 7.30.

7.5.31 SmartMeta

Dieses PlugIn verändert Metadaten eines Dokumentes anhand einer vorher definierten Mapping-Tabelle. Beispielsweise können Benutzereingaben in Schlagwörter für Suchmaschinen umgewandelt werden, ohne dass eine manuelle Eingabe des Benutzers notwendig ist. Dieses PlugIn benötigt keine weitere Konfiguration.

Name	SmartMeta
Gruppierung	Steuerung
PlugIn Name	SmartMeta.pm

Tabelle 7.31.

7.5.32 Workflow einbinden

7.5.32 Workflow einbinden

Mit diesem PlugIn können Sie komplexe Workflows in Module unterteilen. Diese Module funktionieren ähnlich wie Codeincludes und können in mehreren Haupt-Workflows eingesetzt werden.

Der Inhalt wird beim Start des Hauptworkflows eingelesen, so dass der Ablauf genau so ist, als ob die einzelnen Schritte direkt im Hauptworkflow eingebunden wären.

In den PlugIn-Einstellungen wählen Sie aus, welcher Workflow geladen werden soll.

Dies geschieht über die Option *Eingang des Workflow einbinden*.

Dazu ein Beispiel:

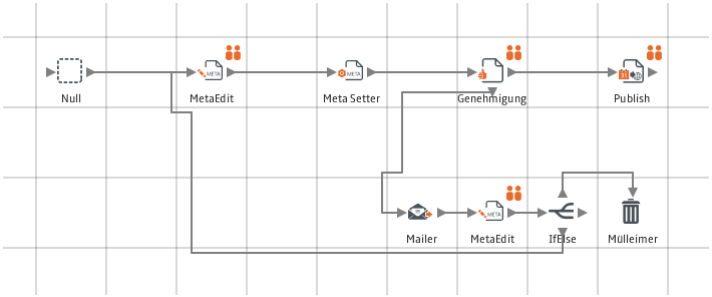


Abb. 7.23: Beispiel-Workflow ohne "Workflow einbinden"

Im oben stehenden Workflow muss ein Dokument vor seiner Veröffentlichung z.B. vom Chefredakteur genehmigt werden.

Falls das Dokument nicht genehmigt wird, kann der Chefredakteur noch zwischen zwei Möglichkeiten wählen. Entweder muss das Dokument korrigiert oder komplett neu erstellt werden. Im ersten Fall wird es zurück in den Bearbeitungs-Schritt geleitet, im zweiten gelöscht. Außerdem wird der zuständige Autor per Email benachrichtigt.

Der komplette Zweig zur Behandlung nicht genehmigter Dokumente soll nun in einen Unter-Workflow ausgegliedert werden.

Dazu wird zunächst ein eigener Workflow angelegt, der die Schritte aus diesem Zweig enthält. Die folgende Abbildung zeigt den Aufbau dieses Workflows:

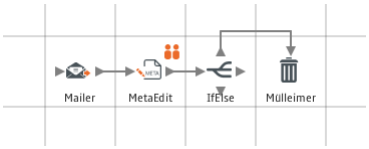


Abb. 7.24: Workflow einbinden

Im Haupt-Workflow wird der entsprechende Zweig entfernt und durch ein PlugIn ersetzt:

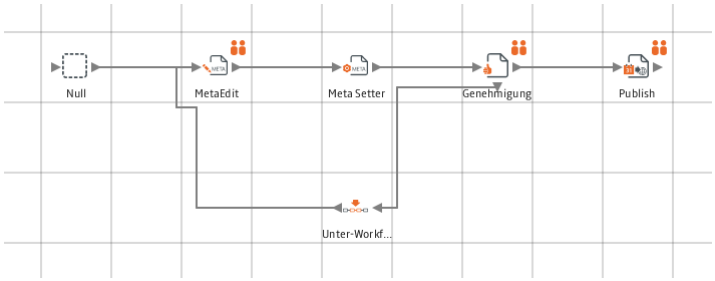


Abb. 7.25: Der abgeänderte Haupt-Workflow

In der PlugIn-Konfiguration wird festgelegt, welcher Workflow in den Haupt-Workflow geladen werden soll.

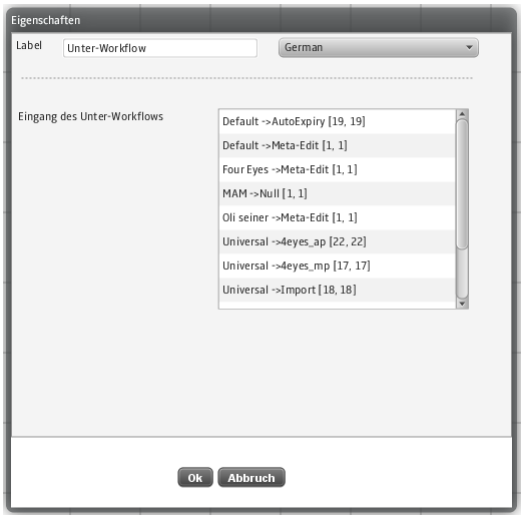


Abb. 7.26: Konfiguration des "Workflow einbinden"

Name	Workflow einbinden
Gruppierung	Steuerung
PlugIn Name	SubWorkflow.pm

Tabelle 7.32.

7.5.33 Asset hochladen

Dieses PlugIn kommt nur im Zusammenhang mit Asset-Dokumenten zum Einsatz. Es steuert die Speicherung der mit einem Asset-Template erfassten Metainformationen sowie die Übertragung der Asset-Datei von Ihrem lokalen zum **imperia** -Entwicklungssystem. Es zählt zu den semi-interaktiven PlugIns, da Sie beim Mehrfachupload nicht für jedes Asset einen eigenen Workflow-Schritt ausführen müssen, sondern alle Assets in einem Durchgang hochladen. Folgende Konfigurationsmöglichkeiten stehen Ihnen zur Verfügung:

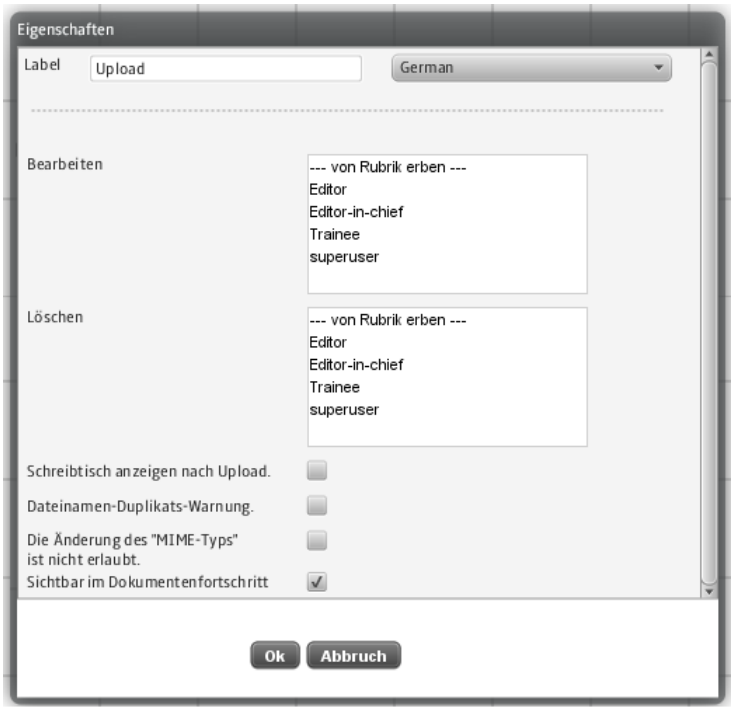


Abb. 7.27: Konfiguration des PlugIns "Asset hochladen"

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Schreibtisch anzeigen nach Asset hochladen

Normalerweise springt der Workflow nach dem "Asset hochladen"-Schritt automatisch weiter zum nachfolgenden Workflow-Schritt. Aktivieren Sie diese Option, wenn Sie möchten, dass stattdessen nach erfolgtem Upload der Schreibtisch erscheint.

Dateinamen-Duplikatswarnung

Wenn Sie diese Option aktivieren, prüft **imperia** vor dem Speichern eines Assets im Upload-Schritt, ob in der betreffenden Rubrik bereits ein Asset mit diesem Dateinamen vorliegt. Ist dies der Fall bleibt der Workflow im Upload-Schritt und es erscheint eine Warnmeldung.

Sie können dann entweder durch Markieren der Option **Überschreiben** das bestehende Asset ersetzen lassen oder einen anderen Dateinamen vergeben.

Hinweis:

Beim Mehrfachupload greift diese Option nicht. Sollten Sie nicht im voraus die Option **Überschreiben** *aktiviert haben, wird dort ein Asset bei Dateinamensgleichheit grundsätzlich nicht hochgeladen.*

Die Änderung des "MIME-Typs" ist nicht erlaubt

Diese Option greift ein, wenn ein bestehendes Asset erneut importiert werden soll. Sie verhindert dabei, dass ein Objekt mit einem anderen Dateiformat als das bestehende hochgeladen wird.

Wenn das Asset zum Beispiel ein JPEG-Bild war, wird mit dem Setzen des Häkchens verhindert, dass ein PNG-Bild hochgeladen wird. Dies dient dazu, dass keine Bilddaten unter einer falschen Endung (hier: PNG-Bild mit der Endung .jpg) in das Asset Management eingespielt werden.

Name	Asset hochladen
Gruppierung	Interaktiv, Assets, Erfassung
PlugIn Name	Upload.pm

Tabelle 7.33.

7.5.34 Wiedervorlagedatum

Dokumente nach einer bestimmten Zeit wieder zur Überarbeitung auf den Schreibtisch importieren.

It is also possible to specify a resubmit date in a metafile via DATE "resubmit:Wiedervorlage" "EXPIRE" "2013-08-23 13:14" "/aaadocs/expiry/000008" "index.html" "EXPIRE" "2012-07-21 13:14" "/aaadocs/expiry/000007" "index.html" "EXPIRE" "2011-06-20 13:14" "/aaadocs/expiry/000006" "index.html"

Dieses PlugIn benötigt keine weitere Konfiguration.

Name	Wiedervorlagedatum
Gruppierung	Interaktiv, Steuerung, Qualitätssicherung, Erfassung
PlugIn Name	Resubmit.pm

Tabelle 7.34.

7.5.35 Lastverteilung

Dieses PlugIn besitzt drei Ausgänge, von denen es einen zufällig auswählt und das Dokument an das daran anschließende PlugIn weitergibt.

Jedem der Ausgänge kann eine Gewichtung zugewiesen werden, durch die die Verteilung der Dokumente auf die einzelnen Ausgänge beeinflusst werden kann. Zur Berechnung der Gewichtung für einen Ausgang wird die Summe der drei zugewiesenen Gewichtungen durch den dem betreffenden Ausgang zugewiesenen Wert geteilt.

Wird beispielsweise dem nördlichen Ausgang eine Gewichtung von 4, dem östlichen eine Gewichtung von 10 und dem südlichen eine Gewichtung von 6 zugewiesen, werden die Dokumente wie folgt verteilt:

- 20% nach Norden
- 50% nach Osten
- 30% nach Süden

Anstelle von Zahlen können die Parameter auch vom Template-Prozessor berechnet werden, beispielsweise `<!--XX-time_to_spare-->`. Sollte ein Parameter leer oder `0` sein, werden diese Parameter behandelt, als hätten sie den Wert `1`, um eine Division durch `0` zu vermeiden.

Dieses PlugIn scheint auf den ersten Blick ein reines Spielzeug zu sein, wenn Sie jedoch einige hundert Dokumente von einem externen Content-Provider in **imperia** importieren, die bearbeitet oder genehmigt werden müssen, kann mit Hilfe dieses PlugIns die Last automatisch auf mehrere Autoren verteilt werden.

Konfiguration

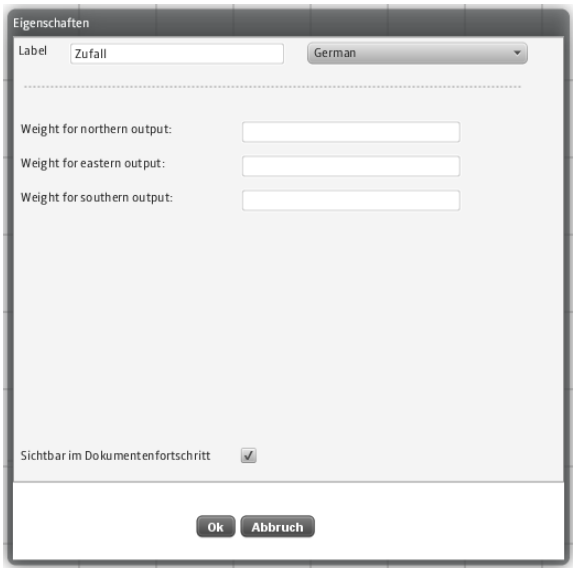


Abb. 7.28: Konfiguration des PlugIns Lastverteilung

Name	Lastverteilung
Gruppierung	Steuerung
PlugIn Name	Random.pm

Tabelle 7.35.

Kapitel 8. Menü-Management

In diesem Kapitel finden Sie Informationen über die Verwaltung der **imperia** -Menüs und der Zugriffsberechtigungen der Rollen auf Menüs und Untermenüs. Für jede im System eingerichtete Rolle kann eine eigene Kombination von Menüpunkten konfiguriert werden, die dann für alle Benutzer, die diese Rolle annehmen, gilt.

Es können auch eigene Untermenüs angelegt und mit Icons versehen werden. Zu einem Menü gehören folgende Teile:

- ein Symbol,
- Zugriffsrechte für Rollen,
- ein Link zum Aufrufen des entsprechenden Skripts,
- ein Linktext, der zum Öffnen des Menüs dient und
- eine Beschreibung.

Alle Arbeiten an den Menüs werden im Dialog **Menü bearbeiten** durchgeführt. Dieser Dialog ist in drei Abschnitte gegliedert:

- Menüabkürzungen,
- Zusätzliche Links,
- Menü-Konfiguration.

Menüabkürzungen

In diesem Abschnitt werden die ID und der Titel des Menüs festgelegt. Wenn Sie das Menü bearbeiten, können Sie die ID nicht ändern. Beim Erstellen oder Bearbeiten von Untermenüs können Sie sowohl die ID als auch den Titel ändern.

Hinweis:

Bei Änderungen an der ID eines bestehenden Untermenüs erzeugt **imperia** *automatisch ein neues Untermenü mit der neuen ID.*

In diesem Bereich sehen Sie außerdem eine Liste aller bereits vergebenen IDs sowie weiterer ebenfalls reservierter Begriffe, die Sie nicht als IDs verwenden dürfen.

Zusätzliche Links

Dieser Abschnitt gibt Ihnen die Möglichkeit, in dem Menü, das Sie bearbeiten, zusätzliche Buttons anzeigen zu lassen, mit denen die Benutzer aus dem Menü heraus bestimmte Funktionen aufrufen können.

Der Abschnitt ist als Tabelle aufgebaut, die aus den Spalten **Symbol**, **Rolle**, **Link-Text**, **Beschreibung** und **Link** besteht.

8.1 Untermenü hinzufügen

In neuen Untermenüs können zum Beispiel die **imperia** -Funktionen zusammengefasst werden, die nur einer bestimmten Rolle zugewiesen wurden. Jeder Benutzer sieht in jedem Menü nur die Menüpunkte, die seiner aktuellen Rolle zugeordnet sind.

Um ein neues Menü zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü hinzufügen**.

   Es öffnet sich der Dialog **Menü bearbeiten**.

2. Tragen Sie im Abschnitt **Menüabkürzungen** eine ID und einen Titel für das neue Menü ein.

Unterhalb der Eingabefelder werden alle IDs aufgelistet, die schon vergeben sind. Diese können Sie nicht verwenden.

3. Nun müssen Sie die einzelnen Menüpunkte erstellen bzw. konfigurieren. Dazu steht der Bereich **Menü-Konfiguration** zur Verfügung.

   Geben Sie zunächst in der ersten Tabellenzeile in der Spalte **Symbol** die Nummer des Symbols an, das für den ersten Menüpunkt verwendet werden soll. In **imperia** stehen ca. 80 Symbole zur Auswahl. Diese sind über eine Select-Box in der Spalte **ICON** auswählbar. Sobald Sie eines der Icons aus der Liste ausgewählt haben, bekommen Sie oberhalb der Select-Box eine Vorschau angezeigt. Vorher sehen Sie dort ein Fragezeichen.

   Wählen Sie dann in der Spalte **Rollen** eine oder mehrere Rollen aus, die auf den Menüpunkt zugreifen dürfen. Markieren Sie dazu die entsprechenden Rollen in der Liste. Mehrere Rollen werden mit der Maus und gedrückter Strg-Taste markiert.

   Tragen Sie in der Spalte **Link** den Link ein, den der Menüpunkt ausführen soll. Zum Beispiel `http://www.imperia.de` oder `/cgi-bin/site_general_settings.pl`.

   Geben Sie in der Spalte **Linktext** den Text ein. Dieser Text erscheint später fett neben dem Symbol. Der Benutzer kann wahlweise durch Anklicken des Symbols oder des Textes den Link ausführen. Tragen Sie in der Spalte **Beschreibung** einen kurzen Text ein, der dem Benutzer die Funktion des Menüpunkts verdeutlichen soll.

   Führen Sie diese Schritte für jeden neuen Menüpunkt aus und verwenden Sie für jeden Menüpunkt eine eigene Tabellenzeile.

4. Klicken Sie auf den Button **Okay**, um das neue Menü zu speichern. Der Button **Abbrechen** verlässt den Dialog, ohne Eingaben zu speichern.

   Neue Menüdateien werden im Verzeichnis `/site/config` abgelegt. Der Dateiname wird aus der Menü-ID und der Endung `.mnu` gebildet.

Das Erstellen eines neuen Menüs ist damit abgeschlossen. Das neue Menü kann nun der Ordnerliste hinzugefügt werden. Lesen Sie hierzu das folgende Kapitel.

8.1.1 Untermenü einem Ordner hinzufügen

Jedes **imperia** -Menü enthält drei nicht belegte Menüpunkte, die Sie verwenden können, um eigene Einträge zu machen. Sollte diese Anzahl nicht ausreichen, speichern Sie zunächst das Menü. Nach dem Speichern stehen erneut drei freie Menüpunkte zur Verfügung.

Um einen neuen Menüpunkt in die Linkliste des Menüs aufzunehmen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** den Menüpunkt **Menü bearbeiten**.

   Es öffnet sich der Dialog **Menü bearbeiten**.

2. Tragen Sie in der Spalte **Symbol** in einer freien Zeile die Symbolnummer ein, die verwendet werden soll.
3. Markieren Sie in der Spalte **Rolle** die Rollen, die zum Zugriff auf das neue Menü berechtigen.
4. Geben Sie nun in der Spalte **Link** folgenden Link ein:

   ```
   /cgi-bin/site_main.pl
   ```

5. Ergänzen Sie diesen Link durch den Aufruf des neuen Menüs:

   ?*Menüname*

   Wenn Sie ein Menü mit der ID *menue1* erzeugt haben, muss der Link wie folgt aussehen:

```
/cgi-bin/site_main.pl?menue1
```

6. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Änderungen zu speichern.

   Sie gelangen in einen Dialog, von dem aus Sie das geänderte Menü über einen Testlink anzeigen lassen können.

7. Mit dem Button **Okay** gelangen Sie zum Menü-Manager, den Sie über den Button **Schließen** verlassen können. Sie gelangen dann automatisch in das Menü.

8.2 Menüs bearbeiten

Mit Hilfe dieser Funktion werden der Aufbau der Menüs und die Zugriffsrechte auf die Menüpunkte eingestellt.

Sie können die im Menü angezeigten Texte verändern, weitere Menüpunkte dem Menü hinzufügen, vorhandene Menüpunkte aus dem Menü entfernen und die Rollen auswählen, die zum Zugriff auf die einzelnen Menüpunkte berechtigen.

8.2.1 Menüpunkt zum Menü hinzufügen

Um einen Menüpunkt hinzuzufügen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Menü bearbeiten**.

   Es öffnet sich der Dialog **Menü bearbeiten**.

2. Ändern Sie in den einzelnen Abschnitten die Angaben und Einstellungen.

   Näheres hierzu erfahren Sie in den folgenden Kapiteln.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Änderungen und Einstellungen zu speichern.

   Der Button **Abbrechen** schließt den Dialog, ohne Änderungen zu speichern.

8.2.2 Menüpunkt einem Untermenü hinzufügen

Um einen Menüpunkt einem Untermenü hinzuzufügen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü bearbeiten**.

   Es öffnet sich eine Liste aller eingerichteten Untermenüs.

2. Wählen Sie das Untermenü, dem Sie einen weiteren Menüpunkt hinzufügen wollen, aus.

   Es öffnet sich der Dialog **Menü bearbeiten** mit den Daten des gewählten Menüs.

3. Tragen Sie in einer freien Zeile die erforderlichen Daten des neuen Menüpunktes ein. Verfahren Sie hier wie in Abschnitt 8.1 **Untermenü hinzufügen** auf Seite 126 beschrieben.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Änderungen zu speichern. Über den Button **Abbrechen** verlassen Sie den Dialog, ohne Änderungen zu speichern.

   Es erscheint ein Hinweis, dass das Menü gespeichert wurde, den Sie ebenfalls mit dem Button **Okay** bestätigen müssen. Sie gelangen daraufhin in den Menü-Manager.

5. Schließen Sie den Menü-Manager mit Hilfe des Buttons **Schließen**. Sie gelangen daraufhin in das geänderte Menü.

8.2.3 Zugriffsrechte für Menüs einstellen

Um die Zugriffsrechte für Menüs einzustellen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü bearbeiten**.

   Es öffnet sich eine Liste aller eingerichteten Untermenüs.

2. Wählen Sie das Untermenü, dem Sie einen weiteren Button hinzufügen wollen, aus.

   Es öffnet sich der Dialog **Menü bearbeiten**, mit den Daten des gewählten Menüs.

3. Markieren Sie nun für jeden Menüpunkt in der Spalte **Rolle** die Rollen, die zum Zugriff berechtigen bzw. heben Sie die vorhandene Markierung für nicht zum Zugriff berechtigte Rollen auf.

   Um eine Markierung aufzuheben, klicken Sie bei gedrückter Strg-Taste auf den markierten Eintrag. Mehrere Markierungen werden mit Hilfe der Strg-Taste getätigt.

8.2.4 Untermenü löschen

Um ein Untermenü zu löschen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü löschen**.

   Es wird eine Liste aller Untermenüs angezeigt.

2. Klicken Sie auf das Menü, das gelöscht werden soll.

   Es erscheint eine Sicherheitsabfrage, die Sie mit **OK** bestätigen müssen. Daraufhin wird das Untermenü gelöscht und eine Bestätigung wird angezeigt.

3. Klicken Sie auf den Button **Okay**, um in das Menü zurückzugelangen.

Hinweis:

Die Standardmenüs von **imperia** *lassen sich nicht löschen. Dies ist nur bei nachträglich erstellten Menüs möglich.*

8.3 Menüstruktur anzeigen

Um die vorhandene Menüstruktur anzeigen zu lassen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Menüstruktur anzeigen**.

   Es wird eine Liste aller Menüs und Untermenüs sowie der dazugehörigen Links angezeigt.

2. Klicken Sie auf den Button **Zurück** am unteren Rand der Liste, um diese Ansicht wieder zu verlassen.

Kapitel 9. MVC i9+

Die Darstellung von **imperia 9** erfolgt konsequent über das Model-View-Controler (MVC) Architekturmuster.

Vor **imperia 9** wurde bereits die eine versionierte Datenhaltung (Model) im Produkt eingeführt, um dieses Konzept vorzubereiten. In **imperia 9** wurde die Ausgabeschicht konsequent in Controller und Views getrennt.

Die Folge der konsequenten Verwendung von Views ist, dass es die Bedienungselemente, wie z.B. Listen oder Bäume, einheitlich dargestellt und bedient werden können.

9.1 Model

9.2 View

9.2.1 Views debuggen

Mit dem SYSTEM_CONF Schalter IMPVIEW_ERRORS können Fehler in den Views propagiert werden. Optionen sind: HTML, JS, STDERR

```
"IMPVIEW_ERRORS" = "HTML,JS,STDERR"
```

HTML : Die Fehlermeldung wird als HTML-Kommentar mit in die Seite geschrieben.

JS : Die Fehlermeldung wird als JS-Alert in der Seite ausgegeben.

STDERR : Die Fehlermeldung wird nach STDOUT geschrieben und wird je nach Konfiguration ins error-log des Webservers oder in das Imperia Error.log geschrieben.

Wird "IMPVIEW_ERRORS" nicht gesetzt, so ist der Default STDERR und die Fehlermeldungen werden in das Logfile geschrieben.

"IMPVIEW_DEVTOOLS" = "1" : Inkludiert einen Data::Dumper in die View, gibt aber noch nichts aus.

"IMPVIEW_CACHE_POLICY" = "timestamp" : Cached die Views und läd sie neu, wenn ihr Timestamp sich geändert hat. Default :immer cachen.

9.3 Controler

9.3.1 Controller-Berechtigungen i9+

Das Gegenstück zu den alten **imperia** (pre9) Skript-Berechtigungen ist der Controller-Berechtigungsbaum in **imperia 9** Das ganze System ähnelt dem Berechtigungssystem vom Verzeichnisstrukturen in einem regulären Dateisystem, vor allem aus dem Benutzer-Sicht.

Versuchen Sie mit Ihrem intern Webserver `site/bin/site_imperia.pl` aufzurufen und loggen Sie sich mit dem Browser als ein **imperia** Superuser unter `http://localhost:2404/imp/system/permissions` ein. Was man nun sieht ist der **imperia** -Controller Baum. Man sieht die Berechtigungen für den Berechtigungsbaum selbst, indem man den Top-Level-Knoten "system" ausklappt und auf "permissions" klickt. Die Beschriftungen der Knoten entsprechen URIs, eigentlich erst der Teil hinter dem führenden `/imp` (oder dem Präfix entsprechend, der in der SYSTEM_CONF Variable DYNAMIC_URI_PREFIX konfiguriert wurde). All diese Conrollerpfade entsprechen einem Perl-Modul (ein Controller) im Dateisystem. Die Zuordnung ergibt sich wie folgt:

Das Basisverzeichnis für alle Controller ist das Verzeichnis `modules/core/Imperia/Controller` in Ihrem site-Verzeichnis. Das erste Zeichen eines jeden Teils des Pfades eines Controllers wird in Großbuchstaben umgewandelt, jeder Unterstrich im Pfad entfernt und das nachfolgende Zeichen nach dem Unterstrich wird kapitalisiert.

Beispielsweise wird der Pfad `/document/foo_bar` in `/Document/FooBar` umgewandelt werden. Dies stellt den Teil des Basispfads dar, der für alle Dateien gültig ist, die im Zusammenhang mit dem Controller stehen.

Die Dateien, die im Zusammenhang mit der Berechtigungsbaum `/System/Permissions` selbst stehen, entsprechen den zuvor eingeführten Regeln `modules/core/Imperia/Controller/System/Permissions.*`.

In den Blättern des Berechrigungsbaums findet man folgende Dateien für jeden Controller:

- Permissions.pm, das Perl-Modul mit dem Controller-Code
- Permssions.icon, eine Icon-Kennung, die durch ein exsitierendes Icon aus dem Verzeichnis "/imperia/images/icons" erweitert wird
- Permssions.docref, eine Kennung für die Dokumentation, die im Verzeichnis unter /imperia/documentation/imperia liegt
- Permissons.description, eine kurze Beschreibung des Controllers
- Permissions.acl, die Access-Control-List (ACL) für diesen Controller

Wenn kein Perl-Modul vorhanden ist, d.h. wenn ein Knoten im Berechtigungsbaum nur ein Verzeichnis ist (z.B. "/document/media"), können alle anderen Dateien in dem Verzeichnis angefügt werden.

Die Access-Control-List eines Controllerpfads (oder site-Area, etc.) kann durch einen Klick auf den Namen im Berechtigungsbaum bearbeitet werden. Die wichtigste Berechtigung ist das Ausführungsrecht der Controller, einfach "access" genannt. Schreibrechte ermöglichen das Bearbeiten der Berechtigung für diesen Knoten im Baum. Leserechte sind nicht sicherheitsrelevant. Mit Leserechten lässt sich nur die Sichtbarkeit eines Knotens im Baum einstellen. Das System erzwingt Leserechte für jeden mit Schreibrechten.

Im Gegensatz zu anderen Berechtigungsbäumen in **imperia** (vor allem der Rubrikenbaum) verwendet der Berechtigungsbaum "wahre" Vererbung für sein Rechtesystem. Wenn ein Knoten im Baum einen übergeordneten Knoten besitzt, kann man einfach dessen Berechtigungen vererben, indem die entsprechende Checkbox gesetzt wird. "Wahre" Vererbung deshalb, weil diese Relation dauerhaft ist. Im Rubrikenbaum zum Beispiel, können die Berechtigungen der übergeordneten Rubrik auch "vererbt" werden, allderings werden spätere Berechtigungsänderungen des Elternknotens vom Kindknoten nicht übernommen. Internes "vererben" wird durch das Löschen der ACL-Datei des Controllerpfads realisiert.

Ein weiteres neues Feature des Berechtigungsbaum sind die sogenannten "free rides". Man kann den Zugriff auf bestimmte Bereiche des Systems für jeden angemeldeten Benutzer erlauben, indem man erneut die entsprechende Checkboxen setzt. Obwohl es theoretisch möglich ist, verwehrt die Benutzeroberfläche den öffentlichen Schreibzugriff auf Controllerbereiche. Intern wird Öffentlichkeit durch die ungültge Rollen-ID "*" implementiert.

Wie funktioniert das Perl seitig? Wann werden Zugriffe überprüft? Wie kann festgestellt werden, ob ein bestimmter Benutzer Zugriff auf einen bestimmten URI hat? Interessanterweise führt die Basisklasse aller Controller `Imperia::Controller` keine Authentifizierung oder Autorisierung durch. Deshalb sollten alle **imperia** -Controller von der Basisklasse `Imperia::Lib::Controller` abgeleitet werden (erben), welche wiederum von `Imperia::Controller` erbt. `Imperia::Lib::Controller` implementiert die _auto()-Methode, die immer vor den eigentlichen Handler-Methoden go() bzw. dispatch() aufgerufen wird. Wenn "undef" zurückgegeben wird, wird die Verarbeitung der Anfrage gestoppt. Die _auto ()-Methode von `Imperia::Lib::Controller` führt zwei AA-Dinge aus:

Zunächst wird der Benutzer **authentifiziert**, indem das Cookie, das Kennwort oder ein anderer konfigurierter Token überprüft wird. Danach wird die **Autorisierung** des Benutzers überprüft, indem festgestellt wird, ob der Benutzer Zugriff auf den geforderten Bereich hat.

Die Autorisierungsbfrage erfolgt durch das Aufrufen der Methode _allowed(), die im `Imperia::Controller` implementiert ist. Diese Methode ist eine Klassemethode, so dass der Controller zur Überprüfung der Berechtigung nicht instanziiert werden muss. Drei optionale Argumente können übergeben werden: Das erste ist der Pfad (z.B. `/system/permissions`) welche als Standardwert den aktuellen Controllerpfad hat. Das zweite optionale Argument ist die Art des Zugriffs, die standardmäßig auf "x" für ausführen gestzt ist, falls nicht Schreibberechtigungen (mit "w" für write) oder Leseberechtigungen (mit "v" für view) gefordert sind. Das letzte optionale Argument ist die Benutzer-ID, die standardmäßig dem aktuellen Benutzer entspricht. Genau genommen ist das letzte Argument eine Rollen-ID, d.h. entweder eine (negative) Gruppen-ID oder eine (positive)) Benutzer-ID.

Kapitel 10. Referenzen und Zielsysteme

10.1 Global Metainfo

Mit dem Media Asset Management wurde das Konstrukt des `globale Metainfo-Objekts` eingeführt. Darin werden Inhalte zum Dokument abgelegt die das Dokument, unabhängig von seiner Version oder dem Workflow, gegenüber anderen Dokumenten repräsentieren.

Daraus folgt, dass es für ein Dokument immer auch genau ein `globales Metainfo-Objekt` gibt. Dieses Objekt besitzt keine Versionierung und kann am Workflow vorbei bearbeitet werden. Die Daten des zugehörigen Dokuments sind davon jedoch nur indirekt betroffen. Diese können nur unter Einhaltung des Workflows bearbeitet werden.

Im globalen Metainfo-Objekt werden die die folgenden Meta-Informationen gespeichert, soweit diese vorhanden sind:

```
__imperia_ac
__imperia_mam_unique
__imperia_asset_actions
__imperia_ac_variants
__imperia_ac_prevfile
__imperia_created
__imperia_mam_spool_original
__imperia_mime_content_length
__imperia_mime_content_type width height
__imperia_old_mdb_datatype
comment1
comment2
copyright
dpi
playlength
sellprice
server
viewprice
```

Zusätzlich werden alle Felder die in der System-Conf Variablen `GLOBAL_METAKEYS` aufgeführt sind in das globale Metainfo-Objekt übernommen.

Wird ein Dokument dessen `globales Metainfo-Objekt` geändert wurde importiert, so werden die Metainformationen im Dokument überschrieben. Ausnahme ist das Metafeld „Template“. Dieses Metafeld wird nicht überschrieben.

Beim „reparsen“ werden weder die Werte vom Dokument ins `globale Metainfo-Objekt` übernommen, noch werden Werte aus dem `globalen Metainfo-Objekt` in das Dokument übernommen.

Wichtig:

Die globalen Metainformationen ermöglichen es am Workflow vorbei Daten zu ändern, die dann beim re-importieren aktiv werden. Dies wiederspricht dem **imperia** *-Philosophie, dass Daten nur im Workflow verändert werden können.*

Hauptgrund für das globale Metainfo-Objekt sind die Varianten der Assets. Sollte es eine einfache Möglichkeit geben Varianten in (kleinen) Workflows zu erstellen, so könnte man ggf. auch ohne das Konzept der globalen Metainformationen auskommen.

Diese (kleinen) Workflows könnten dann auch zur Eingabe eingabe eines einzelnen Datums, wie z.B. des Titels verwendet werden. Bei entsprechender grafischer Aufbereitung würde der Anwnder garnicht merken, dass er sich in einem Workflow-Schritt befindet.

Ohne ein globales Metainfo-Objekt müsste dann durch andere Mechanismen gespeichert werden, welche Metainformationen in den Index gelangen.

Bei den Varianten der Assets wäre dann zu klären welche Auswirkungen das Speichern der Varinaten im „normalen“ Metainfo-Objekt mit sich bringt.

10.2 Referenzen

Unter Referenzen werden in **imperia** URL-Links innerhalb der Metainformationen des Dokuments verstanden die auf ein anderes **imperia** Dokument verweisen. Damit Referenzen umfassend und vollständig erfasst und gespeichert werden können, ist es notwendig genau zu wissen welches Dokument auf welchem Zielsystem vorhanden ist. Ab **imperia 9** werden die Informationen welches Dokument auf welches Zielsystem publiziert bzw. von dort gelöscht wurde detaillierter gespeichert.

Ablaufbeispiel: Ein Dokument A wird erstellt. Im Bearbeiten-Schritt wird z.B. über das Linktool oder das iWE ein Link auf ein anderes **imperia** -Dokument (B) im Dokument A erzeugt. Dazu wird in diesem Beispiel der Titel des Dokuments B als Linktext und die Url als Linkadresse verwendet. Beim Beenden des Workflows werden die in den Metainformationen referenzierten Dokumente erkannt und in ein Metafeld im global Metainfo des Dokuments gespeichert. Zusätzlich wird diese Information in einer Datentabelle (SQLite-Tabelle bei File65) abgelegt. Wenn im Dokument B nun der Titel oder ein anderes Datum geändert wird, so wirkt sich dies aktuell nicht auf das Dokument A aus.

Hinweis:

Eine Reaktion auf diesen Sachverhalt wäre durch aus möglich, die durchzuführende Aktion müsste dann an den Anwendungsfall angepasst werden.

Werden Assets im MAM verschoben so wird in den Dokumeten die verschobene Assets enthalten der jeweilige Pfad zum Asset aktualisiert und das Dokument wird neu freigeschaltet, wobei die Referenzen aktualisiert werden. Befindet sich das Dokument noch im Workflow, so werden die Pfade angepasst aber die Referenzen werden nicht direkt aktualisiert, da dies in den nächsten Workflowschritten automatisch erfolgt.

Wichtig:

Was ist wenn sich der Dateiname ändert?

Soll nun Dokument B gelöscht werden, so wird an Hand der Referenztabelle geprüft, ob andere Dokumente Dokument B referenziert haben. In diesem Fall würde man, untern anderem, angezeigt bekommen, dass Dokument A Dokument B referenziert und das Dokument B aus diesem Grund nicht gelöscht werden kann.

10.2.1 Referenzen parsen

Ab **imperia 9** werden die Referenzen eines Dokuments beim Beenden des Workflows oder beim Einsatz eines entsprechenden Workflow Plug-Ins (ABC1) geprüft. Dabei werden alle Metainformation des Dokuments überprüft und gefundene Referenzen in das `globale Metainfo-Objekt` und die Referenztabelle übernommen.

Hinweis:

Da die Referenzen erst beim Beenden des Dokuments identifiziert werden, werden referenzierte Dokument innerhalb des Workflows nicht als Referenz betrachtet.

Hinweis:

Beim parsen der Metainformationen wird die Metainformation des Dokuments überprüft, nicht die Metainformationen die für die Copy-Pages relevant sind.

Bei der Überprüfung werden alle Links innerhalb der Metainformationen berücksichtigt. Links die nur im Template enthalten sind können ggf. über den Linkcheck geprüft werden, haben aber keinen Einfluss auf die Referenzen. Zeigt ein gefundener Link auf eine aktuelles **imperia** -Dokument oder zeigt der Link auf ein ehemaliges **imperia** -Dokument so wird dieser Link in die Referenzen aufgenommen. Alle anderen Links werden ignoriert und können ggf. über den Linkcheck geprüft werden.

Wichtig:

Die Bestimmung ob ein Link auf ein ehemaliges **imperia** *Dokument zeigt wird auf Grund folgender Informationen bestimmt. WELCHE?*

Wichtig:

Funktioniert das parsen der Referenzen auch bei den Varianten der Assets?

10.2.2 Referenzen darstellen

Gelangt ein Dokument nach der Bestimmung der Referenzen durch das Workflow Plug-In ABC1 in die Darstellung der Referenzen ABC2, so werden in diesem Schritt fehlende Referenzen angezeigt. Nun bestehen folgende Möglichkeiten

- das referenzierte Dokument re-importieren
- letzten Bearbeiter des referenzierten Dokuments informieren und nach 'Wiederbelebung' fragen
- das Dokument bearbeiten und ein anders Dokument referenzieren

In Löschworkflows ist eine umgekehrte Funktionalität notwendig. Durch das Plug-In ABC3 werden nun alle Dokument angezeigt, die dieses Dokument referenzieren. Daraus lassen sich dann Aktionen wie

- referenzierendes Dokument importieren
- letzten Bearbeiter des referenzierenden Dokuments informieren
- das Dokument nicht löschen sondern wieder publizieren

ableiten.

Neben der Darstellung der Referenzen sollten in dem Plug-In nicht nur fehlende Referenzen, sondern auch fehlende Links geprüft und angezeigt werden. Die Linkpüfung sollte optional im Plug-In sein. Während die Referenzen nur auf Basis der Metainformationen berechnet werden, wird die Linkprüfung nur auf Grundlage des HTMLs durchgeführt.

Mögliche Ausgabe

```
Das Dokument "Im Westen nichts Neues" (/12/153/4553) referenziert die folgenden
Dokumente:
                                  Status            Aktion
"Sonne.jpg"  (/15/223)           verfügbar
"Westen.png" (/15/224)           verfügbar
"news.pdf"   (/16/77/1023)       gelöscht          [import]   [mail]

Bis nicht alle referenzierten Dokumente verfügbar sind, können Sie das Dokument
nicht abschließen. Bis dahin können Sie das Dokument weiter [Bearbeiten] und
ggf. andere Objekte auswählen oder sie veranlassen, dass die gelöschten
Dokumente wieder aktiviert werden.


Der Linkcheck der Seiten ihres Dokuments ergab:

Für die Seite /westen/neu/inde.html.de

                                  Status             Aktion
"/impressum.jpg"             nicht verfügbar        [] ignorieren


Für die Seite /westen/neu/inde.html.en

                                  Status             Aktion
"/imprint.jpg"               nicht verfügbar       [] ignorieren
```

10.2.3 Brechen von Referenzen i9

Referenzen können in **imperia 9** nur noch vom superuser gebgrochen werden.

Unter 'Brechen einer Referenz' wird folgendes Verstanden: Ein Dokument vewendet ein Bild z.B.: 'Hotel.jpg' und ist auf dem Livesystem publiziert. Nun stellt der Autor fest, dass das Bild die Rechte anderer verletzt und möchte es sofort unter allen Umständen von der Website entfernen. Dazu konnte er im MAM das entsprechende Bild auswählen und auf Löschen klicken. Im folgenden Dialog wurde ihm dann angezeigt, welche Dokumente dieses Bild verwenden. Hat er an dieser Stelle bestätigt, so wurde das Bild gelöscht und das Dokument hatte einen gebrochen Link. Um diese gebrochen Links weitgehend zu vermeiden, ist es in **imperia 9** nur noch für den superuser möglich Referenzen zu brechen.

Wie gehts richtig?

Der richtige Weg für das Problem ist, alle Dokumente die das Bild verwenden zu importieren und an deren Stelle ein anderes Bild auszuwählen.

Was passiert wenn der superuser eine Referenz bricht?

Für die Erklärung der Folgen müssen Sie wissen wie in **imperia 9** Referenzen erkannt werden und welche Bedingungen erfüllt sein müssen.

Hinweis:

Ein String ist dann eine gültige Referenz, wenn

- *- dieser String wie ein Pfad aussieht (/aa/bb)*
- *- dieser Sting in einem Metafeld des Dokuments steht*
- *- der Pfad im String auf ein existierendes Dokument zeigt*

Bricht also der superuser eine Referenz bei einem Dokument, so wird das referenzierte Objekt gelöscht. Im Dokument wo dieses Objekt vormals verwendet wurde verbleibt der String in dem Metafeld, aber da dieses Objekt nicht mehr exisitert, ist dieser Sting keine gültige Referenz mehr. d.h. Nach dem Löschen des Objektes hat das Dokument keine gebrochenen Refernzen mehr.

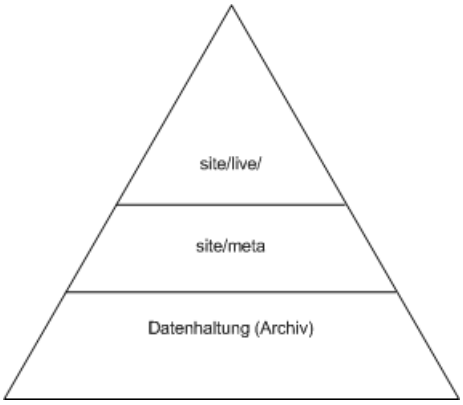
10.3 Zielsysteme

Die Metainformation des Basissystems werden wie auch schon in **imperia 8** im Verzeichnis `site/meta` gespeichert. Die Metainformationen für die weiteren Zielsysteme werden ab **imperia 9** im Verzeichnis `site/live` gespeichert, wobei für jedes Zielsystem ein eigenes Unterverzeichnis angelegt wird.

Um die Verwendung der Zielsysteme in **imperia 9** zu betrachten muss man den Erstellungszyklus eines Dokuments genauer betrachten:

- Das Dokument wird im Workflow erstellt und beendet.
- Eine Datei-Version (üblicherweise eine HTML-Datei) wird unter dem `HTDOCS` Verzeichnis erstellt und eine binäre Datei, in der die Metainformationen enthalten sind, wird in `site/meta` erstellt.
- Bei der Publikation auf die Zielsysteme werden die beiden (bei Copyseite auch mehr als zwei) Dateien auf die entsprechenden Zielsysteme kopiert.

Die Daten im ersten Schritt werden im Archiv gehalten. Da nicht alle Dokumente den Workflow verlassen, enthält der zweite Schritt nur einen Teil der archivierten Dokumente. Und da nicht zwangsläufig alle erstellten Dokumente auf allen Zielsystemen sind, enthalten die Zielsysteme nur einen Teil der erzeugten Dokumente.



Im Umkehrschluss ergibt sich, dass Dokumente die auf einem Livesystem vorhanden sind auch auf dem Developsystem (als Datei unter `HTDOCS` und `site/meta` vorhanden sein müssen. Und Dokumente die auf dem Developsystem vorhanden sind müssen zwangsläufig auch in der Datenhaltung (also im Archiv) vorhanden sein.

Zur technischen Umsetzung dieses Sachverhalts wird die Freischaltliste in **imperia 9** sowohl Schreibaktionen (Dateien) als auch Löschaktionen (Benachrichtigungen) an die Zielssysteme übertragen. Grundsätzlich geht Imperia davon aus, dass alle Dateien auf alle Zielsysteme übertagen werden. Ausnahme ist das Einschränken der Daten durch die Freischalttrigger.

In der Freischaltliste werden nun alle Einträge für jeden Server eingetragen. Dies betrfft das Schreiben der Dateien ebenso wie das Löschen. Wird eine Datei auf ein Zielsystem kopiert, so wird die Metainformation zu diesem Dokument nach `site/live/N/` geschrieben. Gleiches erfolgt analog beim Löschen eines Dokuments.

Die Freischaltliste ist eine Art interne ToDo-Liste die die Aktionen enthält, die durchgeführt werden müssen um das Livesystem an das Devsystem anzugleichen.

Hinweis:

ZUERST WURDE FOLGENDES FESTGELEGT: Eine Bearbeitung z.B. durch das Löschen von Einträgen in der Freischaltliste ist nie erlaubt, da die Systeme dadurch nicht mehr den gleichen Bestand beinhalten würden und die Referenzen somit sinnlos würden. DIESE ENTSCHEIDUNG WURDE WIEDER GESTRICHEN. ES IST NUN DOCH MÖGLICH DELETE-EINTRÄGE ZU LÖSCHEN BZW. SIE ZU ÜBERSCHREIBEN.

Hinweis:

Löscheinträge werden solange in die Freischaltliste ingetragen, bis sie auf allen Livesystemen aktiv wurden. Dazu wird die Löschinformation für alle zu löschenden Varianten je Server in das global Metainfo-Objekt des Dokuments eingetragen.

Ist die Freischaltliste komplett abgearbeitet, so existiert auf den Livesystemen kein Dokument, dass nicht auch auf dem Developsystem existiert. Und es existiert kein Dokument auf dem Developsystem, dass nicht auch in der Datenhaltung vorhanden ist.

Hinweis:

Einzige Ausnahme ist der Zeitbereich bis die Freischaltliste mit den Schreib- und Löschaktionen komplett abgearbeitet wurde. In dieser Zeit können potentiell Dokumente auf dem Developsystem bereits gelöscht worden sein, während sie auf dem Livesystem erst durch die noch anstehende Löschaktion entfernt werden.

Diese Grundlagen sind die Voraussetzung für konsistente Referenzen im System. Aus diesen Grundlagen ergeben sich folgende Implikationen für das Löschen von Dokumenten:

- Ein Dokument kann nur dann aus der Datenhaltung (incl. Archiv) gelöscht werden, wenn dieses Dokument auch vom Developsystem (als Datei unter `HTDOCS` und `site/meta`) gelöscht wird.

- Ein Dokument kann nur dann von Developsystem gelöscht werden, wenn dieses Dokument auch von alle Livesysteme (als Datei unter `HTDOCS` und `site/meta`) gelöscht wird.
- Damit der letzt Punkt sichergestellt werden kann, dürfen die Freischalttrigger nie auf die Löschaktivitäten angewendet werden. Ansonsten könnte ein Freischalttrigger das publizieren einer Löschanweisung verhindern und ein Livesystem würde mehr enthalten als das Developsystem bzw. die Datenhaltung. Das darf im Sinne der Referenzen nicht sein. Aus diesem Grund ignorieren die Freischalttrigger die Löschanweisungen in der Freischaltliste.

Imperia geht bei getrennten Zielsystemen, z.B. bei der Steuerung durch Freischalttrigger, immer davon aus, dass die gleiche Dokumenten-Version auf die Zielsysteme übertragen wird. Unterschiedliche Versionen auf den Livesystemen stellen aus Sicht der Referenzen einen Fehler dar. Man kann diese Abweichung vom Standard in Projekten akzeptieren, muss sich aber darüber im Klaren sein, dass die Referenzen dann nicht im vollen Umfang funktionieren können.

10.3.1 Zielsystem wiederherstellen

OPTIONALE FUNKTION

Mit den besthenden Möglichkeiten in **imperia 9** kann man jederzeit den Stand auf einem Livesystem wiederherstellen in dem man alle Dokumente in `/site/live/N/` neu publiziert und dabei den Transfer-Cache ignoriert.

Zur Wiederherstellung könnten zwei unterschiedliche Varianten verwendet werden. 1. Die re-publikation der bestehenden HTML-Seiten auf Grundlage der in `/site/live/N/` vorhandenen Dokumente. 2. Ein reparsen der in `/site/live/N/` vorhandenen Dokumente.

10.3.2 Beispiele zum Freischalten

Beispiel 1:

Ein Dokument mit der NodeId /1 wird erzeugt in den Sprachen DE und EN es entstehen die Dateien `/index.html.de` und `/index.html.en` im Dateisystem unter `HTDOCS`. Gleichzeitig wird für jede Datei ein Eintrag in der Freischaltliste erstellt, dass dieses Dokument auf das Zielsystem zu schreiben (W) ist, erstellt.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/1      /index.html  DE EN               /1:W:/index.html.de
                                         /1:W:/index.html.en
```

Ein zweites Dokument mit der NodeId /2 wird nur in der Sprache EN erstellt. Es entsteht die Datei `/index.html.en` im Dateisystem unter `HTDOCS`. Diese hat nun die dort von Dokument /1 erstellte Version überschrieben. Also überschreibt auch die englische Version von Dokument /2 den Eintrag in der Freischaltliste. Im Dateisystem wurde die Datei `/index.html.de` nicht verändert, also bleibt auch der Eintrag in der Freischaltliste erhalten.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/2      /index.html  EN                   /1:W:/index.html.de
                                         /2:W:/index.html.en              *x)
```

Es erfolgt das Freischalten. Die Dateien `/index.html.de` (basierend auf Dokument /1) und `/index.html.en` (basierend auf Dokument /2) werden freigeschaltet.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste

Freischalten                             <leer>
```

Nun wird das Dokument /1 reimportiert und ohne Änderung beendet. Es entstehen die Dateien `/index.html.de` und `/index.html.en` im Dateisystem. Diese werden genau so in die Freischaltliste eingetragen.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/1      /index.html  DE EN               /1:W:/index.html.de
                                         /1:W:/index.html.en              *x)
```

Es erfolgt erst mal kein Freischalten und das Dokument /1 wird erneut bearbeitet, wobei aber die Sprachvariante EN entfernt wird. Beim beenden wird nun nur noch die deutsche Seite ins Dateisystem geschrieben. Die englische Seite wird aus dem Dateisystem gelöscht. Analog dazu wird in der Freischaltliste die deutsche Datei zum Schreiben und die englische Datei zum Löschen (D) eingetragen.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/1      /index.html  DE                   /1:W:/index.html.de
                                         /1:D:/index.html.en
```

Es erfolgt ein erneutes Freischalten. Die Datei `/index.html.de` (basierend auf Dokument /1) wird geschrieben `/index.html.en` (basierend auf Dokument /1) wird gelöscht. Es ist der gleiche Stand wie auf dem Developsystem hergestellt.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste

Freischalten                             <leer>
```

Beispiel 2:

Ein Dokument mit der NodeId /1 wird erzeugt in den Sprachen DE und EN es entstehen die Dateien `/index.html.de` und `/index.html.en` im Dateisystem unter `HTDOCS`. Gleichzeitig wird für jede Datei ein Eintrag in der Freischaltliste erstellt, dass dieses Dokument auf das Zielsystem zu schreiben (W) ist, erstellt.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/1      /index.html  DE EN               /1:W:/index.html.de
                                         /1:W:/index.html.en
```

Es wird wieder ein zweites Dokument mit der NodeId /2 nur in englisch erstellt. Es entsteht die Datei `/index.html.en` im Dateisystem unter `HTDOCS`. Diese hat nun die dort von Dokument /1 erstellte Version überschrieben. Also überschreibt auch die englische Version von Dokument /2 den Eintrag in der Freischaltliste. Im Dateisystem wurde die Datei `/index.html.de` nicht verändert, also bleibt auch der Eintrag in der Freischaltliste erhalten.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/2      /index.html  EN                   /1:W:/index.html.de
                                          /2:W:/index.html.en              *x)
```

Es erfolgt das Freischalten. Die Dateien `/index.html.de` (basierend auf Dokument /1) und `/index.html.en` (basierend auf Dokument /2) werden freigeschaltet.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste

Freischalten                             <leer>
```

Nun wird das Dokument /1 reimportiert und es wird die englische Variante entfernt. Beim beenden entsteht nun die Datei `/index.html.de` diese wird auch so zum Schreiben (W) in die Freischaltliste eingetragen. Die englische Datei wurde gelöscht und müsste zum Löschen in die Freischaltliste eingetragen werden. Dies geschieht nicht, da die englische Version Dokument /2 gehört.

```
NodeId  Dateiname    Sprachvarianten      Freischaltliste
/1      /index.html  DE                   /1:W:/index.html.de
                                          [/2:W:/index.html.en] skipped    *x)
```

Hinweis:

VORSCHLAG

*Einführung einer SYSTEM-CONF Variablen z.B.: STRICT_PATHS. Wird diese Variable auf einen wahren Wert gesetzt, so wird immer eine Warnung in die Logdatei geschrieben, wenn ein Dokument mit einer anderen NodeId eine bestehende URL überschreibt. (Alle *x) Stellen.)*

Systeme in denen einzelne Seiten (z.B. die Homepage) immer wieder neu erstellt werden, sollten dann mit dem DocSelctor arbeiten und immer wieder die alte Version des Dokuments bearbeiten. (Das hält auch die Zahl der Dokumente und Versionen geringer.)

Der Vorteil ist, dass das System bei Überschreibungen warnt.

Kapitel 11. Media Asset Management (MAM)

Das Media Asset Management, kurz MAM, dient der Verwaltung unstrukturierter Daten und ermöglicht prozessorientiertes Dokumenten-Management.

Hinweis:

Aus Kompatibilitätsgründen ist die bisherige Mediendatenbank zunächst weiterhin in **imperia** *enthalten. Um diese anstatt des Media Asset Managements zu verwenden, setzen Sie die in der Datei* `site/config/system.conf` *die Variable* `MAM_DISABLE` *auf den Wert 1. Hinweise zum Gebrauch der bisherigen Mediendatenbank entnehmen Sie bitte der Dokumentation zu* **imperia** *8.*

Media-Assets sind imperia -Dokumente

Unstrukturierter Content wie Grafiken, Sound- und Video-Dateien sowie PDF- und Office-Dokumente lassen sich als **imperia** -Dokumente im System verwalten. Die unstrukturierten Dokumente werden so zum integralen Bestandteil des prozessorientierten Content-Managements. Der Upload erfolgt Workflow-basiert. Hierfür stehen Workflow-Plug-Ins zur Verfügung. Weitere Plug-Ins zur Manipulation der Metadaten lassen sich hinzufügen. Da Assets **imperia** -Dokumente sind, versioniert und archiviert sie das System automatisch. So haben Sie jederzeit Zugriff auf ältere Versionen eines Assets und auf die zugehörige Änderungshistorie.

Media-Assets sind frei strukturierbar

Im Media Asset Management sind Sie hinsichtlich der Strukturierung Ihrer Medienobjekte sehr flexibel. Sie können beispielsweise die Rubrikenstruktur Ihres Contents im Media Asset Management ohne Redundanzen abbilden. Bei den Verzeichnissen im MAM handelt es sich um **imperia** -Rubriken, über deren Einstellungen Sie die Zugriffsberechtigungen für die darin befindlichen Assets festlegen.

Media-Assets und Linkmanagement

Im Media Asset Management können Sie nachvollziehen, ob Referenzen auf ein Asset bestehen. Referenzierte Assets lassen sich nur nach einer zusätzlichen Bestätigung löschen.

11.1 Mit Asset-Rubriken arbeiten

Eine Asset-Rubrik besitzt alle Eigenschaften einer herkömmlichen Dokumenten-Rubrik in **imperia** . Sie können in den Einstellungen einer Asset-Rubrik Metainformationen hinterlegen, die für alle darin angelegten Assets verfügbar sind. In einer Rubrik erzeugte Assets folgen einem definierten Workflow und erhalten per Default den Dateinamen, den Sie in den Rubrikeinstellungen festlegen. Für den Upload von Assets und das Anlegen von Asset-Rubriken können Sie Templates definieren. Lesen Sie Abschnitt 11.1.1 **Asset-Rubrik anlegen** auf Seite 140 für weitere Informationen zu den Eigenschaften von Asset-Rubriken. Einzelne Mediendateien finden Sie auch im Archiv über die zugehörige Rubrik.

Für jede Asset-Rubrik gibt es auch ein Verzeichnis im Dateisystem, in dem die Dokumente, sprich Assets, dieser Rubrik abgelegt sind. Das ist aber lediglich eine von vielen Eigenschaften der Mediendateien und nicht mehr ausschlaggebend für deren Strukturierung und Verwaltung.

11.1.1 Asset-Rubrik anlegen

Markieren Sie zunächst im Verzeichnisbaum des Media Asset Managements die Rubrik, in der Sie eine neue Unterrubrik anlegen möchten. Danach rufen Sie die Funktion **Datei**->**Rubrik erzeugen** auf.

Um eine neue Rubrik für Assets zu erstellen, benötigen Sie eine Rolle, die in der Ausgangsrubrik unter "Rollen, die diese Rubrik verändern dürfen" eingetragen ist (siehe Abschnitt 5.1 **Rubriken-Eigenschaften bestimmen** auf Seite 69). Ist dies nicht der Fall, erscheinen die Menüpunkte ausgegraut.

Wenn Sie kein eigenes Template für das Anlegen von Rubriken erstellt haben, gelangen Sie in den Dialog "**Neue Rubrik**". Ab hier ähnelt der Ablauf dem des Anlegens neuer Dokument-Rubriken (siehe Abschnitt 3.1.1 **Neue Rubrik anlegen** auf Seite 40 und Abschnitt 5.1 **Rubriken-Eigenschaften bestimmen** auf Seite 69). Legen Sie folgende Einstellungen für die neue Asset-Rubrik fest:

Rubriken-Informationen, Workflow und Template

Name
Der Name der Rubrik. Unter diesem Namen finden Sie die Rubrik im Media Asset Management, in der Rubriken-Verwaltung und im Archiv.

Beschreibung
Tragen Sie in diesem Feld einen optionalen Beschreibungstext für die Rubrik ein. Dieser erschient im Archiv als Tooltip für den Info-Button einer Rubrik.

Verzeichnis
Legen Sie das Verzeichnis bzw. den Pfad fest, in dem die Assets in dieser Rubrik auf dem Server liegen.

Dateiname
Hier geben Sie den Default-Dateinamen für in der Rubrik erstellte Dokumente an. Für den Upload von Assets ist dieser Eintrag unerheblich, da Sie den Dateinamen des Assets im Verlauf des Upload-Workflows eintragen.

Nur Unterrubriken
Wählen Sie diese Option, wenn Sie in der Rubrik keine Assets, sondern nur Unterrubriken anlegen möchten.

Dokumente und Unterrubriken
Mit dieser Option können Sie in der Rubrik sowohl Unterrubriken als auch Assets anlegen.

Assetspeicherung erlauben?
Diese Option ermöglicht das Anlegen von Assets in der betreffenden Rubrik. Sie müssen Sie nur bei der Hauptrubrik des Media Asset Managements markieren. Bei allen darunter liegenden Asset-Rubriken ist sie automatisch aktiviert.

Workflow-Auswahl
Wählen Sie hier den Workflow **MAM**, der den Ablauf des Uploads von Assets regelt.

Metadatei
Da der Workflow für den Asset-Upload keinen Meta-Edit-Schritt beinhaltet, ist keine Metadatei notwendig. Tragen Sie in dieses Feld **dummy** ein.

Template
Wählen Sie hier das Template für den Asset-Upload aus. Sie können hierfür das Standard-Template verwenden oder ein eigenes Template erstellen. Lesen Sie hierzu den Abschnitt "**Datei-Upload (Media Asset Management)** " in der Syntax-Referenz für **imperia** -Templates im Programmierhandbuch.

Zugriffsrechte

In diesem Bereich bestimmen Sie, welche Rollen in welcher Form auf die Rubrik und die darin befindlichen Assets zugreifen dürfen.

Hinweis:

Die hier vorgenommenen Einstellungen sind äquivalent zu den globalen und den verzeichnisbezogenen Berechtigungen für die Mediendatenbank aus älteren **imperia** *-Versionen.*

Rollen, die diese Rubrik lesen dürfen
Rollen, die diese Berechtigung haben, sehen den Inhalt der betreffenden Asset-Rubrik. Sie ermöglicht die Verwendung der darin befindlichen Assets in Dokumenten. Außerdem muss ein Benutzer auch dann Leserecht für eine Rubrik haben, wenn er Zugriff auf deren Unterrubriken haben soll.

Rollen, die Dokumente in dieser Rubrik anlegen dürfen
Diese Berechtigung ermöglicht das Anlegen von Assets in dieser Rubrik.

Rollen, die Dokumente in dieser Rubrik löschen dürfen
Mit dieser Berechtigung ist das Löschen von Assets und Unterrubriken in der betreffenden Rubrik erlaubt.

Rollen, die diese Rubrik verändern dürfen
Benutzer mit Rollen, die hier eingetragen sind, dürfen Unterrubriken erstellen und die Einstellungen der Rubrik ändern.

Speichern Sie die neu angelegte Rubrik durch Klick auf **Okay**. Sie gelangen daraufhin ins Rubriken-Management von **imperia** , wo Sie die soeben erstellte Rubrik sehen können.

11.1.2 Asset-Rubrik bearbeiten

Wählen Sie zunächst im Rubrikenbaum des Media Asset Managements die gewünschte Rubrik aus. Rufen Sie anschließend die Funktion **Datei**->**Rubrik bearbeiten** auf. Sie gelangen zum Dialog "**Rubrik bearbeiten**". Die Einstellungsmöglichkeiten ähneln weitgehend denen des Dialogs "**Neue Rubrik**" (siehe vorhergehender Abschnitt und Abschnitt 5.1 **Rubriken-Eigenschaften bestimmen** auf Seite 69). Zusätzlich haben Sie noch die Möglichkeit, beliebige Metainformationen für die Rubrik zu erfassen. Weitere Einzelheiten zu Rubriken-Metainformationen finden Sie im Abschnitt 3.1.2 **Rubriken bearbeiten** auf Seite 41. Nehmen Sie die gewünschten Einstellungen vor und speichern Sie diese durch Klick auf **Okay**.

11.1.3 Asset-Rubrik löschen

Zum Löschen einer Asset-Rubrik wählen Sie zunächst die gewünschte Rubrik im Rubrikenbaum des Media Asset Managements aus. Danach rufen Sie die Funktion **Datei**->**Rubrik löschen** auf. Sie können nur leere Rubriken löschen. Befinden sich noch Assets und / oder Unterrubriken in einer Rubrik, erhalten Sie bei einem Löschversuch die Fehlermeldung "Rubrik ist nicht leer". Der Löschvorgang wird in diesem Fall abgebrochen. Entfernen Sie zunächst alle Assets bzw. Unterrubriken aus der Rubrik bzw. verschieben Sie diese in eine andere Asset-Rubrik und führen Sie dann den Löschvorgang erneut aus.

11.2 Import von Assets aus einem Ordner oder einer Zip-Datei

Möchten Sie eine größere Anzahl von Assets in das Media Asset Management importieren, so können Sie dies entweder auf der Kommandozeile oder über eine grafische Oberfläche erledigen.

Das Kommandozeilentool ist mächtiger; kleinere Anzahlen von Assets ohne besondere Anforderungen können bequemer über die Oberfläche importiert werden.

11.2.1 Vorarbeiten und Konfiguration

Bevor Sie jedoch erfolgreich importieren können, müssen einige Einstellungen gemacht werden.

Setzen Sie die System.conf-Variable `MAM_GET_UPLOAD_FILENAME` auf den Wert 1. Siehe Abschnitt 17.16.6 **Variable MAM_GET_UPLOAD_FILENAME** auf Seite 228.

Standardmäßig werden Zip-Dateien aus dem Verzeichnis `/spool` importiert. Sie können über die System.conf-Variable `MAM_SPOOL_PATH` aber ein anderes Verzeichnis, das unterhalb des site-Verzeichnisses liegen muss, wählen. Legen Sie zunächst das Verzeichnis, aus dem die Dateien importiert werden sollen, an und sorgen Sie für Leserechte des Webservers in diesem Verzeichnis. Verzeichnisrechte können für das Spool-Verzeichnis und die direkten Unterverzeichnisse über „Spool Chmod“ eingerichtet werden.

Beispiel:

```
"MAM_SPOOL_PATH" = "/mam_import"
```

Mit der Variable `MAM_MAX_SPOOL_SIZE` können Sie die Maximalgröße für den Import von Assets festlegen. Der erwartete Wert der Variablen ist die Byte-Anzahl. Ist der Wert nicht vorhanden, wird keine Größenbeschränkung vorgenommen.

Beispiel, Beschränkung auf 100 MB:

```
"MAM_MAX_SPOOL_SIZE" = "104857600"
```

Durch Einfügen der Variablen `PLUGIN_ARCHIVE_ZIP_BIN` in die Datei `system.conf` können Sie festlegen, welche Bibliothek für das Entpacken verwendet werden soll. Ist die Variable nicht vorhanden oder der Wert leer, wird versucht, das PERL-Modul `Archive::Zip` zu verwenden, falls das installiert ist.

Beispiel:

```
"PLUGIN_ARCHIVE_ZIP_BIN" = "/usr/bin/unzip"
```

Hinweis:

Beachten Sie, dass ZIP-Dateien bestimmte Dateigrößen nicht überschreiten sollten, sonst kommt es beim Entpacken zu Problemen. Bei 32-Bit-Systemen darf die Größe 2 GB nicht überschreiten. Auf 64-Bit-Systemen können ZIP-Dateien bis 4 GB entpackt werden.

11.2.2 Import mit Hilfe des Kommandozeilentools

Das Kommandozeilenskript zum Import wird folgendermaßen aufgerufen: Wechseln Sie in das Basisverzeichnis Ihrer **imperia** -Installation des Entwicklungsservers. Rufen Sie `site/bin/mam_spool_import.pl` mit den passenden Parametern auf. Folgende Parameter sind möglich:

- -f, --file=Dateiname

  Dateiname, relativ zum Spool-Verzeichnis (üblicherweise site/spool, kann aber in der system.conf geändert werden).

  Als Dateiname kann eine einzelne Datei, ein Verzeichnis, das Dateien enthält, oder ein Zip-Archiv angegeben werden. Es können mehrere Optionen -f gesetzt werden. Die Angabe mindestens eines Files ist Voraussetzung für den Programmablauf.

- -c, --catid=CATEGORY_ID

  ID der MAM-Rubrik, in die importiert werden soll. Die Angabe eines Kategorie ist Voraussetzung für den Programmablauf.

- -u, --uid=UID

  imperia -Benutzer-ID des Mitarbeiters, der als Ersteller der MAM-Objekte fungieren soll.

- -e, --extract

  Dateien aus einem Zip-File sollen entpackt werden.

- -l, --follow-cat

  MAM-Kategorien sollen für Verzeichnisse in dem angegebenen Archiv oder Verzeichnis angelegt werden.

- -d, --bydir

  Bei der Zuordnung von MAM-Kategorien zu importierten Kategorien soll der Verzeichnisname der MAM-Kategorie und nicht, wie voreingestellt, der Kategorienname verglichen werden.

- -w, --overwrite

  Assets mit dem gleichen Pfad wie ein importiertes Asset sollen überschrieben werden.

- -b, --background

  Das Skript soll im Hintergrund laufen und Ausgaben in das Logfile geschrieben werden.

- -t, --batch

  Ausgaben des Skriptes sollen statt auf der Konsole in das Logfile geschrieben werden.

- -q, --quit

  Tritt ein Fehler auf, soll der Programmablauf sofort gestoppt werden.

- -h, --help

Ein Hilfetext wird ausgegeben und der Import nicht durchgeführt.

- -v, --verbose

  Ausführlichere Meldungen sollen ausgegeben werden.

Beispiele:

```
mam_spool_import.pl -c /1/15/143 -u 1 -f "company.jpg" -f "agb.zip" -f "logo.png"
```

Die Dateien company.jpg, agb.zip und logo.png werden mit dem Benutzer mit der ID 1 in die MAM-Rubrik /115/143 hochgeladen. Das Zip-Archiv wird nicht entpackt.

```
mam_spool_import.pl -c /12 -u 1 -e -l -f "images.zip"
```

Das Archiv `images.zip` wird entpackt und die enthaltenen Dateien werden in die MAM-Rubrik `/12` bzw. deren Unterrubriken hochgeladen. Dabei werden die in dem Zip-Archiv enthaltenen Verzeichnisse mit den Rubrikennamen des MAM verglichen und die Assets in die vorhandenen Rubriken einsortiert. Nicht bestehende Rubriken werden angelegt.

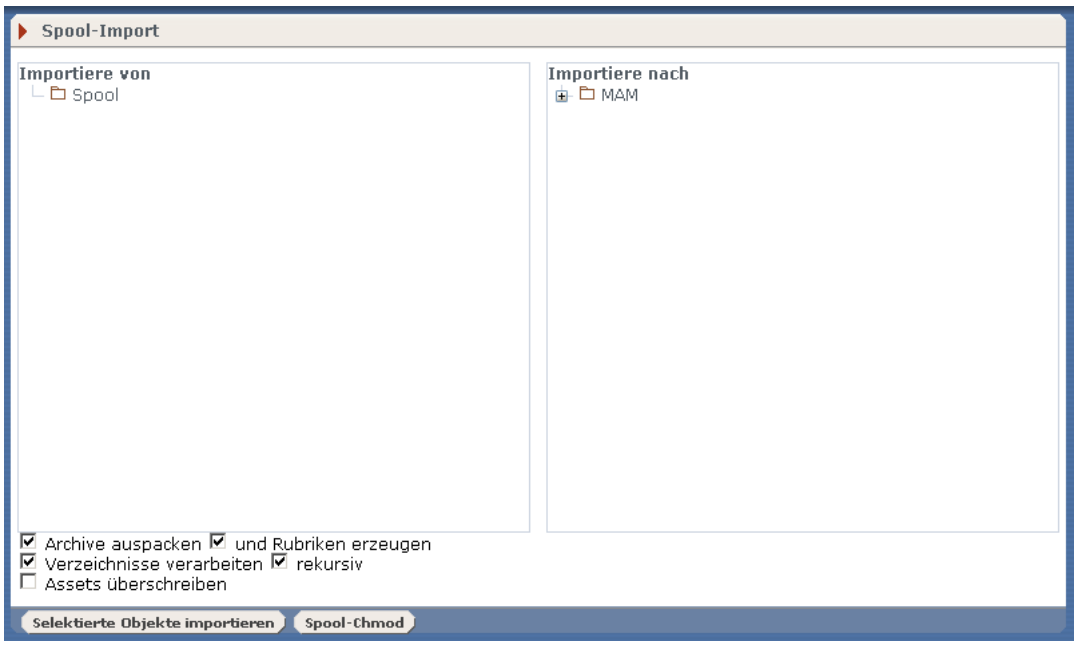
```
mam_spool_import.pl -c /12 -u 1 -l -d -w -f "importverzeichnis"
```

Die in dem Verzeichnis `site/spool/importverzeichnis` enthaltenen Dateien werden in die MAM-Rubrik `/12` bzw. deren Unterrubriken hochgeladen. Dabei werden die in dem Ordner enthaltenen Verzeichnisse mit den Rubriken-Verzeichnisnamen verglichen und die Assets in die vorhandenen Rubriken einsortiert. Nicht bestehende Rubriken werden angelegt. Vorhandene Assets werden gegebenenfalls überschrieben.

11.2.3 Import über die grafische Oberfläche

Die grafische Oberfläche des MAM-Imports rufen Sie wie folgt auf:

`/cgi-bin/site_mam_spool_import.pl`.



11.2.3.1 Vorarbeiten und Konfiguration

Bevor Sie jedoch erfolgreich importieren können, müssen einige Einstellungen gemacht werden.

Setzen Sie die System.conf-Variable `MAM_GET_UPLOAD_FILENAME` auf den Wert 1.

```
"MAM_GET_UPLOAD_FILENAME" = "1"
```

Standardmäßig werden Zip-Dateien aus dem Verzeichnis `/spool` importiert. Sie können über die System.conf-Variable `MAM_SPOOL_PATH` aber ein anderes Verzeichnis, das unterhalb des `site`-Verzeichnisses liegen muss, wählen. Legen Sie zunächst das Verzeichnis, aus dem die Dateien importiert werden sollen, an und sorgen Sie für Leserechte des Webservers in diesem Verzeichnis. Verzeichnisrechte können für das Spool-Verzeichnis und die direkten Unterverzeichnisse über "Spool Chmod" eingerichtet werden.

Beispiel:

```
"MAM_SPOOL_PATH" = "/mam_import"
```

Mit der Variable `MAM_MAX_SPOOL_SIZE` können Sie die Maximalgröße für den Import von Assets festlegen. Der erwartete Wert der Variablen ist die Byte-Anzahl. Ist der Wert nicht vorhanden, wird keine Größenbeschränkung vorgenommen.

Beispiel, Beschränkung auf 100 MB:

```
"MAM_MAX_SPOOL_SIZE" = "104857600"
```

Durch Einfügen der Variablen `PLUGIN_ARCHIVE_ZIP_BIN` in die Datei `system.conf` können Sie festlegen, welche Bibliothek für das Entpacken verwendet werden soll. Ist die Variable nicht vorhanden oder der Wert leer, wird versucht, das PERL-Modul `Archive::Zip` zu verwenden, falls das installiert ist.

Beispiel:

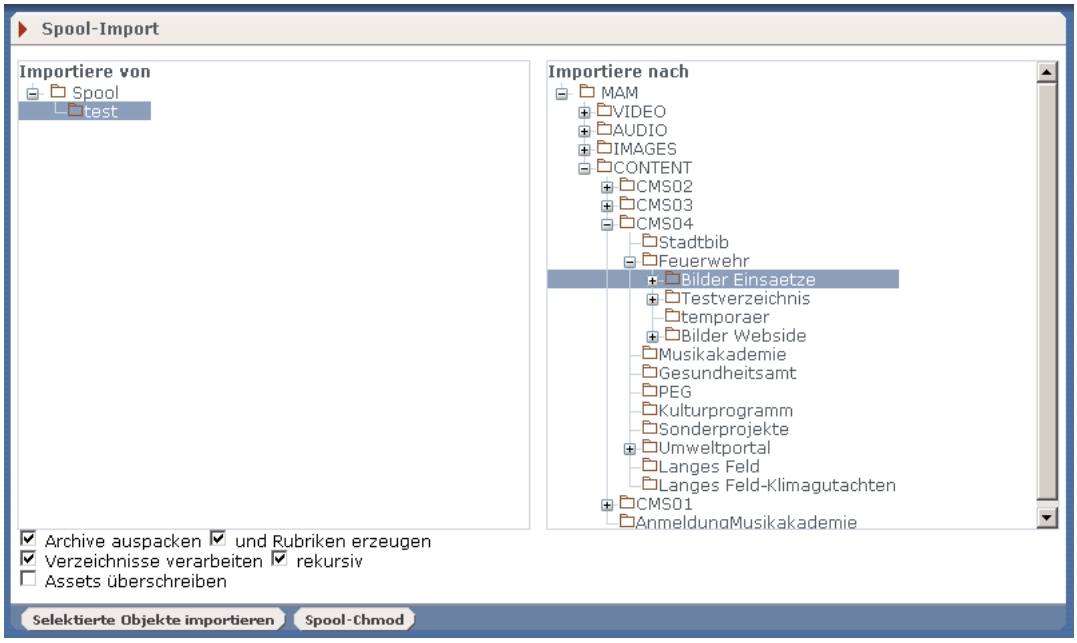
```
"PLUGIN_ARCHIVE_ZIP_BIN" = "/usr/bin/unzip"
```

Hinweis:

Beachten Sie, dass ZIP-Dateien bestimmte Dateigrößen nicht überschreiten sollten, sonst kommt es beim Entpacken zu Problemen. Bei 32-Bit-Systemen darf die Größe 2 GB nicht überschreiten; auf 64-Bit-Systemen können ZIP-Dateien bis 4 GB entpackt werden.

11.2.3.2 Die Web-Oberfläche

Assets aus dem Ordner `site/spool/test` werden in die MAM-Rubrik „Bilder Einsätze“ importiert.



"Archive auspacken" - importierte Archivdateien werden entpackt (derzeit nur Zip-Archive) und die enthaltenen Dateien importiert.

"und Rubriken erzeugen" - beim Import werden MAM-Rubriken für jeden importierten Ordner erzeugt.

"Verzeichnisse verarbeiten" - der Inhalt ganzer Verzeichnisse wird importiert.

"rekursiv" - es werden auch alle Unterordner der gewählten Ordner verarbeitet.

"Assets überschreiben" - Assets, die bereits bestehen, werden durch neu importierte mit dem gleichen Dateinamen überschrieben.

Spool Chmod

Für Autoren, die die Superuser-Rolle haben, wird ein Button "Spool Chmod" angezeigt. Hierüber können Rechte für Importverzeichnisse gesetzt werden. Für jedes Directory (linker Teil des Fensters) können Zugriffsrechte für Rollen (rechts) gesetzt werden. Dies funktioniert in zwei Ebenen, also dem Spoolverzeichnis und einer Verzeichnisebene darunter.

11.3 Konfiguration des Media Asset Managements

Abgesehen von den benutzerabhängig gespeicherten Einstellungen des MAM, die im Benutzerhandbuch beschrieben sind, gibt es einige allgemeingültige Konfigurationsmöglichkeiten für das Media Asset Management. Diese Einstellungen nehmen Sie über Konfigurationsvariablen vor, die Sie in der Datei `site/config/system.conf` notieren.

11.3.1 Globale Suchfilter definieren

Wie im Abschnitt "**Im Media Asset Management suchen**" des Benutzererhandbuches beschrieben, lassen sich Suchfilter abspeichern. Diese gelten allerdings nur für jeden Benutzer individuell. Als Benutzer mit Superuser-Zugang haben Sie jedoch auch die Möglichkeit, definierte Suchfilter so abzuspeichern, dass alle Benutzer Ihres Systems sie anwenden können. Markieren Sie hierfür beim Erstellen oder Bearbeiten eines Filters die Option "**Filter global speichern**" ("Save as global filter"). Das System speichert so erstellte Filter in der Variablen `MAM_SAVED_FILTERS`.

11.3.2 Asset-Metafelder global verfügbar machen

Das Media Asset Management erlaubt die Speicherung beliebiger Metainformationen zu den Assets. Um diese Metainformationen in Dokumenten abrufen zu können, müssen Sie die Metafelder global verfügbar machen. Hierzu verwenden Sie folgende Variable:

```
"GLOBAL_METAKEYS" = "Metafeldname01,Metafeldname02,MetafeldnameNN"
```

11.3.3 Asset-Metafelder in der Listenansicht anzeigen lassen

Globale Asset-Metafelder lassen sich auch als Spalten in der Listenansicht sowie im "Details"-Dialog eines Assets des Media Asset Managements anzeigen. Dies geschieht mit der Variablen `MAM_ADD_USER_FIELDS`. Syntax:

```
"MAM_ADD_USER_FIELDS" = "Metafeldname01:Spaltenüberschrift01,MetafeldnameNN
                            :SpaltenüberschriftNN"
```

Hinweis:

Obenstehender Code ist aus Darstellungsgründen umgebrochen.

11.3.4 Titel-Leiste des MAM-Rubrikenbaums festlegen

Per Default ist der Rubrikenbaum im Media Asset Management mit **Navigation** betitelt. Diesen Titel ändern Sie mit der folgenden Variablen:

```
"MAM_NAV_HEADER" = "Asset-Rubriken"
```

11.3.5 Sortierung des Asset-Rubrikenbaums festlegen

Die Sortierung der Rubriken im Rubrikenbaum des Media Asset Managements erfolgt per Default in der Reihenfolge der Erstellung der Rubriken. Sie können dies auf alphabetische Sortierung umstellen, indem Sie die folgende Variable zur Systemkonfiguration setzen:

```
"MAM_CAT_ORDER_BY_NAME" = "1"
```

11.3.6 Template für die Erstellung von Asset-Rubriken festlegen

Wenn Sie für die Erstellung von Asset-Rubriken ein eigenes Template nutzen, geben Sie dessen Namen mit der Variablen `MAM_CATEGORY_TEMPLATE` an. Notieren Sie den Namen ohne Pfadangabe oder Dateiendung:

```
"MAM_CATEGORY_TEMPLATE" = "Templatename"
```

Templates für die Erstellung von Asset-Rubriken speichern Sie im Verzeichnis `site/templates` ab. Die Syntax für Asset-Rubrik-Templates entspricht der anderer **imperia** -Templates. Ein Beispieltemplate finden Sie unter `site/config/templatecatmeta.htms.sample`.

11.3.7 Standard-Anzahl für Mehrfachupload definieren

Die Option **Datei**->**Mehrfachupload** erlaubt das Anlegen mehrerer Assets in einem Arbeitsgang. Mit der Variablen `MAM_MULTIUPLOAD_SIZE` legen Sie die Standardanzahl hierfür fest.

Beispiel:

```
"MAM_MULTIUPLOAD_SIZE" = "5"
```

11.3.8 Mehrfachupload-Optionen einrichten

Mit der Variablen MAM_MULTIUPLOAD_OPTIONS richten Sie die verfügbaren Optionen für den Mehrfachupload ein. Die Angabe der Optionen erfolgt in Form einer kommaseparierten Liste von Zahlen.

Beispiel:

```
"MAM_MULTIUPLOAD_OPTIONS" = "2,3,4,5,6,7,8,9,10"
```

Hinweis:

Ab einer Anzahl von ca. 15 Assets ist mit steigenden spürbaren Performance-Einbußen zu rechnen. Das ist natürlich nur ein Richtwert. Je nach System-Gegebenheiten kann es dazu auch schon früher kommen.

11.3.9 Linkanzeige der Listenansicht konfigurieren

Mit der Variablen `MAM_LINKS_STYLE` legen Sie die Darstellung der Links in den Spalten der Listen-Ansicht fest.

Die Syntax lautet wie folgt:

```
"MAM_LINKS_STYLE" = "[Option]"
```

Diese drei Optionen stehen zur Auswahl:

ICON
: In den Spalten erscheinen lediglich kleine Icon-Grafiken.

TEXT
: In den Spalten erscheinen ausschließlich Textlinks.

ICON,TEXT
: In den Spalten erscheinen sowohl Icons als auch Text. Diese Einstellung ist der Default.

11.3.10 Asset-Details fixieren

Im Dialog Asset-Details können Benutzer die Metainformationen eines Assets ändern oder Felder hinzufügen und löschen. Die Felder `directory` und `filename` sind per Default fixiert, um die Datenintegrität von Assets und Dokumenten, die darauf verweisen, zu gewährleisten. Um weitere Felder zu fixieren, verwenden Sie die Variable `MAM_READ_ONLY_FIELDS`. Geben Sie die Felder, die nicht editierbar sein sollen, als kommaseparierte Liste an. Dazu ein Beispiel:

```
"MAM_READ_ONLY_FIELDS" = "height,width"
```

Obenstehendes Beispiel fixiert die Felder height und width. Sie können alle Felder aus dem Dialog "Asset Details" fixieren. Groß- und Kleinschreibung ist dabei irrelevant. Allerdings dürfen Sie bei mehreren Feldnamen keine Leerzeichen vor oder nach den Kommas setzen.

11.3.11 Felder verbergen

Wie im Abschnitt "**Benutzeroberfläche des Media Asset Managements einrichten**" des Benutzerhandbuchs beschrieben, lassen sich die Spalten der Listen-Ansicht ein- und ausblenden. Darüber hinaus können Sie Spalten der Listen-Ansicht sowie etwaige äquivalente Felder im Kontextmenü der Iconbox-Ansicht generell verbergen. Hierfür nutzen Sie die Variable `MAM_HIDDEN_FIELDS` mit folgender Syntax:

```
"MAM_HIDDEN_FIELDS" = "Regex, SpaltenID"
```

Tragen Sie die Felder, die Sie verbergen wollen, als kommaseparierte Liste mit oder ohne Leerzeichen als Wert ein. Die Felder können Sie mit der Feld-ID angeben. Alternativ kann die Angabe auch ohne das Präfix *__mam_* erfolgen. Die folgende Tabelle listet die verfügbaren Felder und zugehörigen IDs auf:

Deutscher Spaltenname (Listen-Ansicht)	FeldID
- (Checkbox)	__mam_select
Zoom	__mam_zoom
Icon	__mam_icon
Titel	__mam_title
Dateiname	__mam_filename
Verzeichnis	-
Dateigröße	__mam_filesize
Details	__mam_details
Refererenzen	__mam_references
Löschen	__mam_delete
Erzeugt von	__mam_createdby
Erzeugungsdatum	__mam_createdat
Template	-
Abmessungen	__mam_dimensions
Benutzeraktion	__mam_action
Benutzervarianten	__mam_variants
Archiv	__mam_archive
Import	__mam_reimport

Tabelle 11.1. Verfügbare Felder im Media Asset Management

Hinweis:

Das Feld **Icon** *lässt sich nur in der Listen-Ansicht verbergen.*

11.3.12 Datenhaltung für Binärdateien bestimmen

Damit Assets versionierbar sind, legt **imperia** eine Kopie der Assetdatei an und archiviert diese. Entsprechend der in den Systemeinstellungen gewählten Datenhaltung speichert **imperia** diese Binärdaten im Dateisystem oder als BLOB-Felder in einer Datenbank. Wenn Sie eine SQL-Datenhaltung verwenden und nicht wünschen, dass **imperia** die binären Assetdaten in der Datenbank speichert, setzen Sie in der Datei `site/config/system.conf` die Variable `BINARY_DOCTREE_PLUGIN` auf den Wert *File65*.

11.3.13 Datei-Benennung für Assets konfigurieren

Per Default legt **imperia** neu erzeugte Assets unter dem Dateinamen im System ab, den Sie in den Einstellungen der betreffenden Asset-Rubrik festgelegt haben. Wie bei allen **imperia** -Rubriken bzw. -Dokumenten ist der Standardwert hierfür "`index.html`". Asset-Dokumente stellen allerdings einen Sonderfall dar, da **imperia** hier die Dateiendung automatisch an den jeweiligen MIME-Type der Asset-Datei anpasst. So wird beispielsweise eine PDF-Datei nicht als "`index.html`" abgespeichert, sondern als "`index.pdf`".

Sollte dieses Default-Verfahren zur Dateibenennung nicht Ihren Anforderungen entsprechen, können Sie es mit zwei Variablen beeinflussen.

Hinweis:

Neben den hier beschriebenen Möglichkeiten zur Beeinflussung des Default-Dateinamens von Assets können Sie diesen natürlich auch durch ein entsprechendes Eingabefeld im Upload-Template oder durch Workflow-Plug-Ins ändern. Einzelheiten hierzu entnehmen Sie bitte der Syntaxreferenz im Kapitel **Templates** *des Programmierhandbuchs.*

MAM_GET_UPLOAD_FILENAME

Setzen Sie diese Variable auf *1*, um dem Benutzer beim Upload ein Eingabefeld für den gewünschten Dateinamen zur Verfügung zu stellen, wie es im folgenden Screenshot zu sehen ist:

Abb. 11.1: Upload-Feld mit Möglichkeit zur Eingabe des Dateinamens

Das Eingabefeld ist mit dem Standard-Dateinamen aus den Rubrikeinstellungen vorbelegt. Tragen Sie den gewünschten Dateinamen ein. Alternativ können Sie nur die Vorbelegung aus dem Eingabefeld entfernen, wenn Sie den ursprünglichen Dateinamen auch für das Asset automatisch übernehmen möchten.

Das Eingabefeld für den Dateinamen ist nur beim ersten Workflow-Durchlauf des Assets sichtbar. Bei einem Reimport erscheint es nicht mehr. Außerdem hat es Vorrang vor einem etwa von Ihnen im Upload-Template definierten Formularfeld zur Bestimmung des Dateinamens.

MAM_UNIQUE

Setzen Sie diese Variable auf 1, um sicherzustellen, dass neu erstellte Assets nicht aufgrund zufälliger Namensgleichheit Varianten anderer Assets überschreiben. **imperia** speichert dann alle Assets inklusive der jeweiligen Varianten in einem eigenen Unterverzeichnis der betreffenden Asset-Rubrik.

Hinweis:

Wenn Sie dieses Verhalten nicht global über eine Systemkonfigurationsvariable aktivieren möchten, können Sie alternativ auch die Variable `__imperia_mam_unique` *verwenden. Diese können Sie in den Rubrik-Metainformationen setzen oder auch beispielsweise mit einem Metasetter im Workflow. Beachten Sie dabei, dass Sie die Variable bereits vor dem Upload-Schritt setzen müssen, damit sie wirksam ist.*

DISABLE_8BIT_FILENAMES

Derzeit unterstützt **imperia** nicht die Verwendung von 8-bit-Zeichen in Dateinamen. Aus diesem Grund werden Umlaute, Leerzeichen, Sternchen, das Dollarzeichen usw. in Dateinamen in Unterstriche umgewandelt.

Bisher war es aber möglich, über den MAM-Upload Assets mit derartigen Datei- oder Verzeichnisnamen hochzuladen. Dies führte dann beim Einfügen in HTML-Seiten zu defekten Links. Der Grund für diese Freiheit war, dass **imperia** nicht sicher herausfinden kann, in welchem Format die Zeichen im Betriebssystem kodiert sind.

Setzen Sie die system.conf-Variable `DISABLE_8BIT_FILENAMES` auf den Wert "1" um die Umschreibung von 8-bit-Datei- und Verzeichnisnamen in einen „sicheren“ Zeichensatz beim MAM-Upload zu erzwingen.

```
"DISABLE_8BIT_FILENAMES" = "1"
```

Hinweis:

Verwenden Sie diese Option nicht, wenn Ihr Dateisystem-Zeichensatz UTF-8 ist, es sei denn, Sie wollen wirklich Latin-1-Dateinamen erzwingen.

Um auch beim Massenimport von Assets die korrekte Übernahme von Zeichen wie deutschen Umlauten oder kyrillischen Zeichen in die **imperia** -Oberfläche zu erreichen, muss auch die Variable `MAM_SPOOL_CHARSET` korrekt gesetzt sein.

MAM_SPOOL_CHARSET

Bei der Verwendung des MAM-Massenimports muss der Betriebssystem-Zeichensatz angegeben werden, um eine korrekte Übernahme von Sonderzeichen in die **imperia** -Oberfläche und Logfiles zu erzielen.

Ist das Betriebssystem des **imperia** -Entwicklungsservers Windows, können Sie als Wert "auto" eintragen. **imperia** wird dann den Zeichensatz selbst ermitteln. Voreinstellung für `MAM_SPOOL_CHARSET` ist UTF-8. Bei Angabe eines Leerstrings wird "windows-1252" angenommen. Ist Ihnen bekannt, dass Ihr Betriebssystem-Zeichensatz UTF-8 ist, ist keine Umrechnung nötig. Kommentieren Sie dann `MAM_SPOOL_CHARSET` aus oder löschen Sie die Zeile aus der system.conf.

Hinweis:

Beim Import von Zip-Archiven ist die Umschreibung der Datei- und Verzeichnisnamen nicht möglich, da eine sichere Erkennung des von dem Zip-Tool beim Packen des Archivs verwendeten Zeichensatzes nicht möglich ist. Verwenden Sie daher, wenn möglich, direkt beim Packen des zu importierenden Archives die Zeichenkodierung UTF-8.

11.3.14 Unabhängige Freischaltung von Assets aktivieren

Normalerweise überträgt **imperia** Assets erst dann auf das Zielsystem, wenn andere Dokumente auf sie referenzieren. Wenn Sie Assets unabhängig freischalten möchten, setzen Sie in der Datei `site/config/system.conf` die folgende Variable:

```
"MAM_ENABLE_PUBLISH" = "1"
```

Assets lassen sich dann automatisch und manuell aus der Freischaltliste heraus freischalten. Sie erscheinen außerdem in der Liste der freigeschalteten Dokumente.

11.3.15 Verfügbare Asset-Aktionen nachträglich festlegen

Diese Funktion steht nur in der Listen-Ansicht zur Verfügung. Wenn Sie nachträglich ändern möchten, welche Aktionen mit einem Asset möglich sein sollen, klicken Sie bei dem betreffenden Asset in der Spalte **Aktionen ändern** auf den gleichlautenden Link. Sie gelangen in den Dialog "Asset Aktionen", wo Sie durch Anklicken Aktionen aktivieren bzw. deaktivieren können. Klicken Sie auf "Speichern", um Ihre veränderten Einstellungen abzuspeichern.

11.3.16 Zugriffsrechte für Asset-Rubriken verwalten

Die Zugriffsrechte für Asset-Rubriken stellen Sie über die Rubriken-Verwaltung ein. Diese rufen Sie über die Funktion **Menü**->**Struktur**-> **Rubriken** auf. Die Zuweisung der Berechtigungen können Sie wahlweise über den Dialog **Rubrik bearbeiten** oder das Rubriken-ACL-Werkzeug vornehmen. Die jeweilige Vorgehensweise ist eingehend in Abschnitt 5.1.4 **Rubriken-Zugriff** auf Seite 71 und Abschnitt 3.1.3 **Rubrikenrechte einstellen** auf Seite 43 beschrieben. Daher erfolgt hier nur eine Erklärung der einzelnen Berechtigungen.

Rollen, die diese Rubrik lesen dürfen, bzw. **Lesen (r)**
: Diese Berechtigung benötigt eine Rolle, um Assets aus der betreffenden Rubrik sehen und in einem Dokument verwenden zu können. Soll auf eine Unterrubrik auch schreibender Zugriff möglich sein, müssen Sie auf die übergeordneten Rubriken ebenfalls das Leserecht erteilen.

Rollen, die Dokumente in dieser Rubrik anlegen dürfen, bzw. **Dokumente erzeugen (c)**
: Rollen mit dieser Berechtigung können neue Assets hochladen. Dieses Recht beinhaltet auch das Anlegen von Asset-Varianten und die Änderung an den Metadaten eines Assets.

Rollen, die Dokumente in dieser Rubrik löschen dürfen, bzw. **Löschen (d)**
: Nur Rollen mit dieser Berechtigung können Assets aus einer Rubrik entfernen oder verschieben.

Rollen, die diese Rubrik verändern dürfen, bzw. **Schreiben (w)**
: Mit dieser Berechtigung können Benutzer die Rubriken-Einstellungen modifizieren sowie neue Unterrubriken anlegen. Diese Berechtigung bezieht sich nicht auf die in der Rubrik befindlichen Assets.

11.3.17 Listen-Ansicht konfigurieren

In der Listen-Ansicht des Media Asset Managements ist das Hinzufügen weiterer Spalten möglich. Zusätzlich hinzugefügte Spalten lassen sich auch wieder entfernen. Außerdem können Sie die Reihenfolge der angezeigten Spalten beeinflussen.

11.3.17.1 Neue Spalten hinzufügen und löschen

Um eine neue Spalte in der Listen-Ansicht hinzuzufügen, rufen Sie im MAM die Funktion **Ansicht**->**Spalten anzeigen**->**Spalte hinzufügen...** auf. Daraufhin öffnet sich folgender Dialog:

Abb. 11.2: Der Dialog Neue Spalte

Legen Sie die Einstellungen für die neue Spalte fest und speichern Sie diese durch einen Klick auf **Hinzufügen**.

ID

Tragen Sie hier den Namen des Asset-Metafelds ein, dessen Inhalt in der neuen Spalte angezeigt werden soll.

Name

Tragen Sie hier den Spaltentitel ein.

Sichtbar

Markieren Sie diese Checkbox, wenn die neue Spalte per Default sichtbar sein soll.

Sortierbar

Wenn Sie diese Option aktivieren, können die Benutzer die Listen-Ansicht nach dem Inhalt der neuen Spalte sortieren lassen.

Um eine zusätzlich hinzugefügte Spalte wieder zu entfernen, rufen Sie im MAM die Funktion **Ansicht**->**Spalten anzeigen**->**Anpassen...** auf. Sie gelangen zum Dialog **"Anpassen"**:

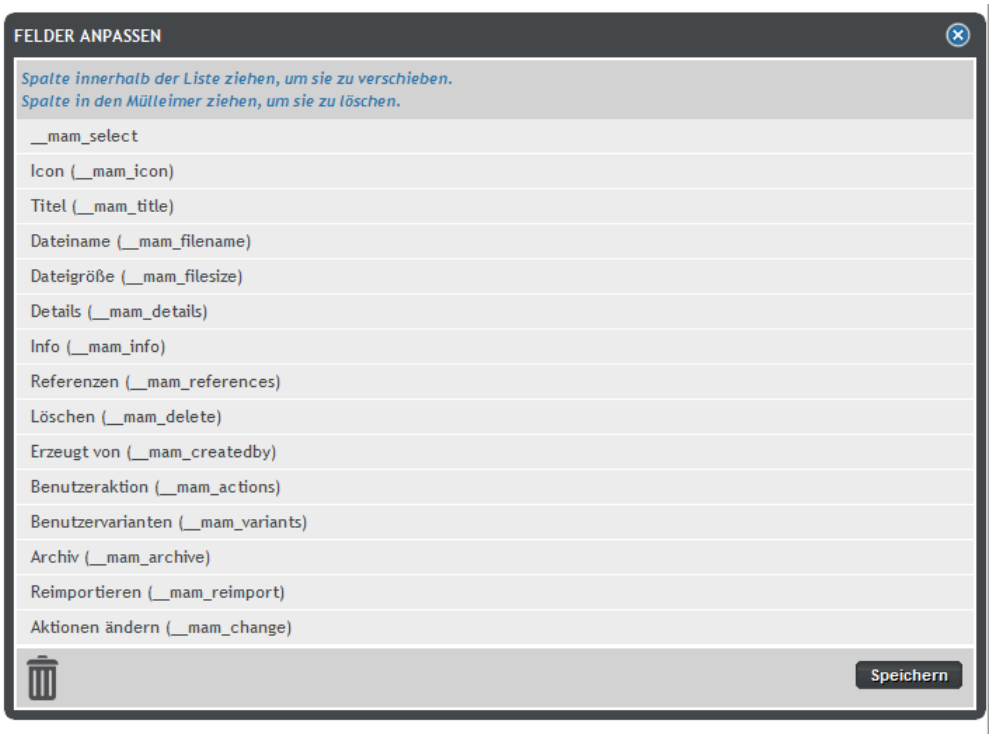


Abb. 11.3: Der Dialog Anpassen

Klicken Sie auf den Listenpunkt vor der Spalte, die Sie entfernen möchten, ziehen Sie diesen mit gedrückter Maustaste über das Papierkorb-Icon im unteren Teil des Dialogfensters und lassen Sie dann die Maustaste los, um die Spalte zu löschen. Anschließend klicken Sie auf speicherm um die Änderung zu Übernhemen und den Dialog schließen.

Hinweis:

Sie können nur nachträglich hinzugefügte Spalten entfernen.

11.3.17.2 Spaltenreihenfolge ändern

Um die Reihenfolge der angezeigten Spalten zu ändern, rufen Sie im MAM die Funktion **Ansicht->Spalten anzeigen->Anpassen...** auf. Sie gelangen zum Dialog "**Anpassen**" (Screenshot siehe oben). Die Spalten erscheinen in von oben nach unten gelesener Reihenfolge der Liste von links nach rechts eingeblendet in der Listen-Ansicht. Klicken Sie auf den Listenpunkt vor einem Spaltennamen, um die betreffende Spalte per Drag-and-Drop in der Liste nach oben oder unten zu verschieben.

11.4 Referenzen

imperia behandelt Assets wie Dokumente. Sie lassen sich gesondert bearbeiten, publizieren und löschen. Daher ist es erforderlich, etwaige Verknüpfungen mit anderen Dokumenten zu verfolgen, um sicherzustellen, dass beispielsweise keine Grafiken gelöscht werden, auf die eines oder mehrere Dokumente referenzieren. **imperia** hält bestehende Referenzen auf Assets nach und warnt Sie, wenn Sie versuchen sollten, ein referenziertes Asset zu löschen. Sie können sich die Referenzen auf ein Asset-Dokument im Media Asset Management anzeigen lassen. Je nach eingestellter MAM-Ansicht gehen Sie dazu wie folgt vor:

Iconbox-Ansicht

Klicken Sie auf das gewünschte Asset, um die zugehörigen Informationen anzuzeigen. Wählen Sie die Option **Referenzen zeigen**. Sie gelangen zum Dialog "**Asset-Referenzen**".

Listen-Ansicht

In der Listenansicht sind die Referenzen auf ein Asset über die Spalte **Referenzen** abrufbar. Eventuell müssen Sie diese Spalte erst anzeigen lassen (lesen Sie hierzu den Abschnitt "**Spalten ein- und ausblenden (nur Listen-Ansicht)**" im Benutzerhandbuch). Bei unreferenzierten Assets erscheint in dieser Spalte der Eintrag **Nicht referenziert.** Bestehen hingegen Referenzen auf ein Asset, erscheint dort der verlinkte Eintrag **Referenzen** mit einem Lupen-Icon. Klicken Sie auf den Link oder das Icon, um den Dialog "**Asset-Referenzen**" zu öffnen.

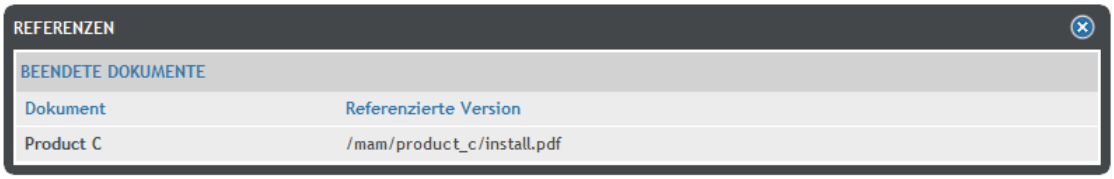


Abb. 11.4: Dialog Asset-Referenzen

Sie erhalten eine Übersicht aller Referenzen auf das Asset und seiner etwaigen Varianten. Beendete und in Bearbeitung befindliche Dokumente sind voneinander getrennt aufgelistet. Die Spalte **Dokument:** enthält die Namen der Dokumente, die auf das Asset zugreifen. Rechts daneben in der Spalte **Referenzierte Version:** ist die jeweils verwendete Variante des Assets angegeben.

Die Namen der Dokumente in der Liste sind mit dem entsprechenden Dokument verlinkt. Klicken Sie auf diesen Link, um eine Vorschau des Dokuments in einem neuen Fenster zu öffnen.

11.5 Einbau in ein Template

Damit ein Benutzer das Media Asset Management in einem Dokument verwenden kann, muss der Template-Code folgende Zeile enthalten:

```
<img src="IMPERIA:defimage1" alt="IMPERIA:default1" height="100" width="100" />
<input name="IMPERIA:default1" type="text" />
```

Hinweis:

Die Angabe von `width` *und* `height` *ist für den Aufruf des Media Asset Managements optional.*

Weitere Einstellungsmöglichkeiten für die Integration des Media Asset Managements in Templates finden Sie im Abschnitt **"Einbindung von Grafikdateien mit dem Media Asset Management"** im Kapitel **"Templates"** im Programmierhandbuch.

11.6 Mehrfachupload mit Flash

Im Unterschied zum Standard-Mehrfachupload ermöglicht diese Variante dem Benutzer, mehrere Dateien innerhalb eines Verzeichnisses auf einmal für den Upload auszuwählen. Da die Implementierung auf Adobe Flash basiert, muss clientseitig eine möglichst aktuelle Version des Flash-Players installiert sein.

11.6.1 Kompatibilität

Der **imperia** -Flashupload ist kompatibel zu folgenden Versionen des Flash-Players:

- Linux Flash v9.0.124.0 (Browser blockiert während des Uploads, siehe **Einschränkungen**)
- Mac Flash v9.0.115.0 und neuer (langsamer Upload bei Version 9.0.124.0)
- Windows Flash v9.0.115.0 und neuer

11.6.2 Einschränkungen

Durch einen Fehler im Flash-Player für Linux blockiert unter Linux der Browser beim Upload von Dateien und reagiert nicht mehr auf Benutzereingaben. Sobald der Upload-Vorgang beendet ist, ist diese Blockade beendet.

11.6.3 Rubrikeneinstellungen

Über die Rubriken-Metavariable `__imperia_swf_upload_template` stellen Sie das Template für den Flash-Multiupload ein. Geben Sie den Dateinamen des Templates ohne Endung oder etwaiges Präfix an. Die Template-Einstellung in den Rubrikeneinstellungen ist für den "normalen" Einzel- bzw. Mehrfachupload reserviert. Die relevanten Anweisungen für Templates für den Flash-Mehrfachupload sind im Abschnitt **Flash-Mehrfachupload (Media Asset Management)** unter **Syntax-Referenz** im Kapitel **Templates** im Programmierhandbuch eingehend beschrieben.

Der Workflow darf keine interaktiven Plug-Ins enthalten. Ausnahme ist hierbei das semi-interaktive Upload-Plug-In aus dem Standard-Workflow MAM. Sobald sich ein interaktives Plug-In im Workflow befindet, bricht der Flash-Upload an dieser Stelle ab.

Wenn Sie im MAM in eine so präparierte Rubrik wechseln, ist über den Menüpunkt **Datei** zusätzlich die Option **Multiupload (Flash)** verfügbar.

11.7 Darstellung der Mime-Typen 19

Im MAM wird für die unterschiedlichen Assetes jeweils ein eigenes Icon zur Darstellung verwendet. Nur bei den webtauglichen Bilden (*.gif, *.jpg, *.png) wird eine Miniaturansicht des Bildes anstelle eines Icons angezeigt.

Default

Application MS Excel

Application MS Word

Application Octet stream

Application PDF

Application MS Powerpoint

Application x-compress

Application x-director

Application x-gtar

Application x-gzip

Application x-shockwave-flash

Application x-zip-compressed

Application zip

	Audio Basic
	Audio Mpeg
	Audio x-pn-realaudio
	Audio x-pn-realaudio-plugin
	Audio x-realaudio
	Audio x-wav
	Audio Default
	Image Default
	Text Default
	Text Html
	Video Quicktime
	Video Default

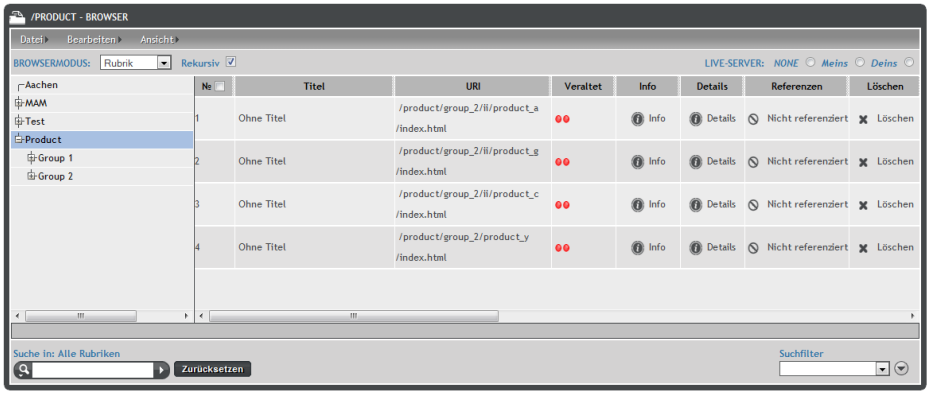
Tabelle 11.2. Darstellung der Mime-Typen

Kapitel 12. Linktool

12.1 Linktool

Mit Hilfe des Linktools können Sie aus bereits erstellten Dokumenten, den Inhalt bestimmter Meta-Variablen an definierte Zielpositionen im aktuell zu bearbeitenden Dokument einfügen.

Das Linktool lehnt sich funktional an seinen Vorgänger, dem Metatool (2) an, wird jedoch ein einem dem MAM ähnlichen Dialog dargestellt.



Ändert sich der Inhalt im verlinkten Dokument später, so wird das aktuelle Dokument nicht automatisch aktualisiert.

Die Oberfläche des Linktools wird über verschiedene Parameter gestaltet. Die Bedienung und der Aufbau des Linktools wird im Kapitel „Linktool“ im Autorenhandbuch beschrieben.

Da das Linktool techisch auf den gleichen Grundlagen wie das MAM basiert, ist auch die Verwendung ähnlich. Ein Link aus dem MAM kann wie folgt implementiert werden:

```
<a href="IMPERIA:MAM:ref_mam_url" >
 <?imperia mam copy
   NAME:ref_mam_text
 ?>» Asset auswählen</a>

<div>Link-URL</div>
  <input name="IMPERIA:ref_mam_url" type="text" size="20" readonly="readonly" />

<div >Linktext</div>
  <input name="IMPERIA:ref_mam_text" type="text" size="20" />
```

Hinweis:

Im Hinblick auf die Referenzen sollte man vorsichtig mit dem expliziten Kopieren der `OBJURL` *umgehen, insbesondere wenn das Eingabefeld nicht schreibgeschützt ist. Es wird intern der Link auf das ausgewählte Dokument in der einen Variablen gespeichert. Wird nun im Eingabefeld die Url verändert, so zeigt diese ggf. auf ein anderes existierendes Dokument. Die Folge ist es werden zwei Referenzen erzeugt, wobei die erste eigentlich nicht benötigt wird. Fazit: Wenn die Url bearbeitet werden kann, sollte man nicht mit*

```
<?imperia mam copy OBJURL:xy>
```

Bzw. im Umkehrschluss: Wenn man Kopien der URL benötigt, sollten diese nicht editierbar sein.

Der entsprechende Aufruf zur Verlinkung eines Dokuments ist:

```
<a class="editHref" href="IMPERIA:DOC:ref_doc_url">
 <?imperia mam copy title:ref_doc_text ?>

 » Linktool öffnen</a>

<div>Link-URL</div>
  <input name="IMPERIA:ref_doc_url" type="text" size="20" readonly="readonly" />

<div >Linktext</div>
  <input name="IMPERIA:ref_doc_text" type="text" size="20" />
```

Warnung:

Bitte beachten: Es können keine Kommentare innerhalb der Processing-Instructions verwendet werden . Kommentare innerhalb des umschliessenden Link-Aufrufs sind ebenso nicht möglich.

12.1.1 Darstellung

Dieser Abschnitt beschreibt die Anweisungen, über die die Darstellung beeinflusst werden kann.

12.1.1.1 Sortierung

Die Sortierung erfolgt über die Anweisung „sortby“ gefolgt von einer der im Linktool verwendeten Spalten.

Hinweis:

Welche Spalten sind das?

Hinweis:

Kann man die getrennt vom MAM definieren und wie?

Hinweis:

Kann man auch nach einer nicht angezeigten Spalte filtern?

```
<?imperia1 mam sortby filesize?>
```

Hinweis:

Wie wird die Sortierrichtung (ASC,DESC) angegeben?

Hinweis:

Wie kann man nach mehr als einer Spalte sortiern?

12.1.2 Prefilter

Über die Prefilter-Anweisungen kann der Template-Entwickler steuern, aus welchem Bereich der Redakteur Dokumente auswählen kann. Diese Einstellung kann vom Redakteur nicht umgangen werden.

Hinweis:

*Wird ein Prefilter gesetzt, so wird dies im Dokumenten- Browser mit einem * markiert. Auf dem Tooltip kann man dann das Feld sehen auf das eingeschränkt wurde.*

12.1.2.1 Metafeld

Ein Filtern mit Hilfe der Prefilter kann mit allen Metafeldern erfolgen, die im globalem Metainfo-Objekt gespeichert sind.

z.B. Ein Perfilter auf das Feld „title“:

```
<?imperia mam prefilter title=~New?>
```

Hinweis:

Die Operatoen entsprechen den Operatoren des MAM. Sie sollten dort oder Zentral dokumentiert sein. PRÜFEN!!

12.1.2.2 Besondere Felder

Neben den Metafeldern kann nach den speziellen Werten

- category
- category_recursive
- directory

gefilterte werden

.

```
<?imperia mam prefilter category = /1/8?>
<?imperia mam prefilter category_recursive = /1/8 ?>
<?imperia mam prefilter directory=/docs/xy ?>
```

Mit category_recursive werden alle Dokumente gefunden, die in der angegebenen Rubrik oder in einer darunterliegenden Rubrik enthalten sind.

.

Hinweis:

Die Prefilter wirken sich (AKTUELL) nur auf die Listendarstellung auf der rechten Seite aus. Der Baum auf der linken Seite wird nicht angepasst. Dies ist besonders bei Filtern auf die Rubrik verwirrend.

12.1.3 Filter

Über die Filter-Anweisungen kann der Template-Entwickler steuern, aus welchem Bereich der Redakteur Dokumente angezeigt bekommt. Der Redakteur kann diese Einstellung ändern und andere Dokumente auswählen.

12.1.3.1 Bildfilter

Filter nach Bildern.

```
<?imperia1 mam filter jpeg?> <
```

Hinweis:

Sollte hier nicht auch in Form einer Zuweisung gearbeitet werden? z.B.

```
<?imperia1 mam filter type=jpeg?> <
```

Hinweis:

Welche weiteren Werte wie „jpeg“ sind vordefiniert?

12.1.3.2 Verzeichnisfilter

Filter nach Verzeichnis.

```
<?imperia1 mam directory /sendung?>
```

Hinweis:

Sollte hier nicht auch in Form einer Zuweisung gearbeitet werden? z.B.

```
<?imperia1 mam directory=~/Sendung?> <
```

Hinweis:

Als Wert auf der rechten Seite sind derzeit auch NodeIds erlaubt. Dies sollte m.E. verhindert werden, da es mit Ziffern in Rubrikennamen kollidieren könnte.

Kapitel 13. Rubriken und Dokumente umstrukturieren i9+

In diesem Kapitel werden alle Funktionen und Anwendungsfälle beschrieben die im Zusammenhang mit dem Verschieben von Rubriken, Dokumenten und Assets zu tun haben. Durch diese Funktion lassen sich bestehende Rubriken- und Verzeichnis-Strukturen leicht an neue gegebenheiten anpassen.

Das Verschieben von Rubriken ist in zwei elementare Teilfunktionen gegliedert. Erstens das „umhängen“ von einer Rubrik unter eine andere und zweitens die Anpassung aller relevanten Pfade. Üblicherweise werden beim Verschieben einer Rubrik auch die Pfade angepasst. Aber beide Teilfunktionen können getrennt von einander ausgeführt werden.

Vom Ablauf ist es so, das zuerst die Auswahl getroffen werden muss, welches Element von wo nach wo bewegt werden soll. Anschliessnd wir die Auswahl angezeigt, und man muss den Vorgang bestätigen. Alle mit dem Verschieben verbundenen Detailaufgaben werden in eine „Todo-Liste“ geschrieben und werden dann nach dieser Liste abgearbeitet. Dieser Lauf kann sehr lange dauern, deshalb besteht die Möglichkeit die Bearbeitung einer „Todo-Liste“ zu unterbrechen.

Der Anstoss eine Rubrik oder Dokumente zu verschieben kann je nach Umgebung und Datenhaltung unter Umständen sehr lange dauern.

As it is not allowd in /structure/new_edit to create two categories with the same names in the same node level, we must also prevent this in the moving.

Hinweis:

Vor dem umstrukturieren muss imperia in den Wartungsmodus versetzt werden um Inkonsistenzen z.B. durch Dokumentenbearbeitung zu verhindern.

Allgemeine Einschränkungen

Das Verschieben von Rubriken ist ein komplexer Vorgang der zum einen sehr lange dauern kann und zum anderen potentielle Gefahren birgt. Aus diesen Gründen gibt es die folgenden Einschränkungen:

- Das Verschieben darf nur von der Rolle „superuser“ ausgeführt werden.
- Es kann immer nur ein Verschiebeprozess laufen. Wenn bereits ein Verschiebeprozess läuft, wird eine Warnung und der Fortschritt dieses Prozesses angezeigt. Erst wenn der Prozess komplett abgearbeitet ist kann ein neuer Verschiebeprozess gestartet werden.
- Beim Verschieben von Dokumenten (ggf. inkl. Rubriken) werden nur Dokumente in Dokumentenrubriken angezeigt. Beim Verschieben von Assets (ggf. inkl. Rubriken) werden nur Assetrubriken angezeigt.
- Ein Verschieben einer Rubrik in eine seiner eigenen Unterrubriken ist nicht möglich.
- Ein Verschieben von Dokumenten unter Assets bzw. umgekehrt ist nicht vorgesehen.
- Momentan ist es nicht möglich eine Unterubrik auf die oberste Ebene zu verschieben.

13.1 Rubriken umstrukturieren

Es werden keine Dateien bewegt. Nur "Umhängen"

Grundsätzlich wird in **imperia** ziwschen Rubriken unterschieden die für Assets vorbereitet sind (__imperia_ac_enabled) und Rubriken die üblicherweise für Dokumente verwendet werden. Im folgenden werden diese der einfachheit halber Asset-Rubriken und Dokument-Rubriken genannt.

Wird eine Asset-Rubrik in unter eine Dokumenten-Rubrik verschoben, so wird aus der ehemaligen Asset-Rubrik (samt Unterrubriken) eine Dokumenten-Rubrik. D.h. __imperia_ac_enabled wird auf 0 gesetzt.

Wird eine Dokumenten-Rubrik in unter eine Asset-Rubrik verschoben, so wird aus der ehemaligen Dokumenten-Rubrik (samt Unterrubriken) eine Asset-Rubrik. D.h. __imperia_ac_enabled wird auf 1 gesetzt.

Im folgenden die Beschreibung für Dokumente, dies gilt analog für Assets. Einzige Ausnahme ist, dass Asset-Varianten im Dialog nicht angezeigt werden, da sie, im Gegensatz zu Copy-Seiten nicht einzeln verschoben werden können.

13.1.1 Rubriken mit Dokumenten umstrukturieren

Nachdem man sich für die Umstrukturierung für Dokumente oder Assets (im weiteren Verlauf wird nur von Dokumenten gesprochen) entschieden hat, gelangt man zur einer Baumansicht die Rubriken und ihre dazugehörigen Dokumente darstellt. Auf linken Seite wählt man die zu Verschiebenden Dokumente/Rubriken aus. Eine Mehrfachselektion ist nicht möglich. Auf rechten Seite kann die Zielrubrik ausgewählt werden.

Im zweiten Dialogschritt werden detailliertere Information sowie Optionsmöglichkeiten zur Umstrukturierung angezeigt.

"Ich möchte zu diesem Zeitpunkt keinen Pfad bearbeiten." Wird diese Option gewählt, schließt sich der Vorschlags Block und es werden keine automatischen Pfadanpasungen vorgenommen nach der Umstrukturierung.

"Existierende Dateien überschreiben" Liegen an den neu berechneten Pfaden andere Dateien als direkt von der Umstrukturierung betroffenen Dokumente, so wird der Prozess angehalten bis diese Dateien manuell entfernt wurden. Wird der diese Option akiviert, so werden alle Dateien ueberschrieben. Copy Pages von direkt von der Umstrukturierung betroffenen Dokumente werden immer überschrieben.

"Verzeichnis der Quellrubrik" Verzeichnis das in der Quellrubrik eingetragen ist.

"Verzeichnis der Zielrubrik" Verzeichnis das in der Zielrubrik eingetragen ist.

"Zielverzeichnis für die zu verschiebende Rubrik" Neu errechnetes Verzeichnis aus Quell- und Zielrubrik.

"Die Verzeichnisse der Rubriken, welche mit dem Verzeichnis der Quellrubrik übereinstimmen, aktualisieren" Der gesamt Rubrikenbaum wird durchsucht, entspricht ein gefundenes Verzeichnis dem Quellverzeichnis, so wird diese entsprechend der Vorschläge angepasst.

"Nur Verzeichnisse von Kindrubriken aktualisieren" Diese Option schränkt die vorherige auf Unterrubriken der zu verschiebenden Rubrik ein.

"Nur die Pfade von Dokumente anpassen, welche zur verschobenen Rubrik gehören" Nur die Pfade der Dokumente die zu der verschobenen Rubrik gehoeren werden gemäß der Vorschläge oben angepasst. Ist diese Option nicht gesetzt, werden die Pfade aller Dokumente angepasst die sich im Verzeichnis der Quellrubrik befinden.

13.1.2 Rubriken mit Assets umstrukturieren

Es gelten die gleichen Bediungungen wie im vorherigem Abschnitte beschrieben.

13.2 Verzeichnissse und Dateien umstrukturieren

Es werden Verzeichnisse und/oder Dateien bewegt bzw. umbenannt. Die Rubrikenstruktur wird nicht geändert.

Im folgenden die Beschreibung für Verzeichnisse. , dies gilt analog für Assets.

13.2.1 Verzeichnisse verschieben

Es können einzelene sowie mehrere Dateien und Verzeichnisse verschoben werden. Es ist nicht möglich eine Datei oder Verzeichnis auf die Root ebene zu verschieben, es ist allerdings möglich durch die Umbennung des Verzeichnisses auf "/" dieses Verschieben zu erreichen.

Im abgebildeten Beispiel wird ein Verzeichnis unterhalb von sport nach lokal verschoben.

"Existierende Dateien überschreiben" Liegen an den neu berechneten Pfaden andere Dateien als direkt von der Umstrukturierung betroffenen Dokumente, so wird der Prozess angehalten bis diese Dateien manuell entfernt wurden. Wird der diese Option akiviert, so werden alle Dateien ueberschrieben. Copy Pages von direkt von der Umstrukturierung betroffenen Dokumente werden immer überschrieben.

"Quellverzeichnis" Verzeichnis das verschoben wird.

"Zielverzeichnis" Verzeichnis in dass das Quellverzeichnis verschoben wird.

"Neues Verzeichnis" Neuer Pfad des verschobenen Verzeichnisses.

"Die Verzeichnisse der Rubriken, welche mit dem Quellverzeichnis übereinstimmen, aktualisieren." Werden in Rubrikeneinstellungen Pfade gefunden, die mit dem Pfad des des zu verschiebenenden Verzeichnis übereinstimmen, so werden die Rubrikeneinstellungen angepasst.

13.2.2 Verzeichnisse umbenennen

In diesem Abschnitt besteht die Möglichkeit entweder nur den Namen eines bestimmten Teil des Verzeichniss zu ändern oder den ganzen Pfad zu ändern.

Nach einem Klick auf auf das Verzeichniss öffnet sich ein Dialog in dem man nur ein Teil des Verzeichnis umbenennen kann.

Betrachtet man das obige Beispiel, so wird aus dem Verzeichnis "/nachrichten/lokal/000980" nach dem umbenennen "/nachrichten/koeln/000980"

Wird der Stift rechts angeklick, so öffnet sich ein Dialog in dem man das ganze Verzeichnis ändern kann.

Nach entsprechender Auswahl werden im zweiten Schritt die gleichen Optionen wie bei "Verzeichnisse verschieben" angezeigt nur das hier "Zielverzeichnis" und "Neues Verzeichnis" leer sind.

13.3 Interne Abläufe

In diesem Abschnitt sollen die Details erwähnt werden, die intern bei einer Umstrukturierung von statten gehen. Dadurch soll das technische Verständniss gestärkt und ein Gefühl für Abhängigkeiten und Laufzeiten gegeben werden.

Wie schon weiter vorne beschreiben, werden die Umstrukturierungen in zwei Schritten durchgführt. Im ersten Schritt wird eine ToDo-Liste erstellt, die dann im zweiten Schritt abgearbeitet wird.

1. Schritt erstellen der ToDo-Liste:

- Scan über den kompletten Rubrikenbaum ob Pfade angepasst werden müssen.
- Neue Rubriken Node-Ids berechnen(nur bei Rubriken umstrukturieren)
- Für jedes betroffene Dokument
  - Neue Node-Id errechnen(nur bei Rubriken umstrukturieren)
  - Alle Referenzen (Links auf dieses Dokument) sammeln
  - Alle Grafiker-Objekte sammeln

Ist die Liste zusammengestellt, so wird eine Zusammenfassung angezeigt wieviele Objekte verändert werden um ein ungefähre Abschätzung der Laufzeit zu ermoeglichen. Man kann hierbei von ca. 1 Sekunde pro Objekt ausgehen.

2.Schritt abarbeiten der ToDo-Liste:

Optionale Elemente die bei der Konfiguration ausgelassen werden können sind mit einem * gekennzeichnet.

- Rubrik in der Datenhaltung "umhängen" (neu berechnete Node-Ids zuweisen)
- Alle betroffenen Dokumente laden, Meta-Informationen anpassen, speichern für
  - Version aus SiteMeta

  - Letzte Version aus dem Archiv
  - ggf. die Version aus dem Workflow
  - betroffene Copy Pages
- Alle betroffenen Dokumente reparsen
- Alle betroffenen Referenten laden, Meta-Informationen anpassen, speichern für
  - Version aus SiteMeta
  - Letzte Version aus dem Archiv
  - ggf. die Version aus dem Workflow
  - betroffene Copy Pages
- Alle betroffenen Referenten reparsen
- Alle Grafiker-Objekte explizit an der alten Position löschen.
- Vorschläge für manuelle Pfadanpassungen darstellen
  - In Template
  - In Flexmodulen
  - In Slotmodule
  - In Includes
  - In der Dynamic.conf
  - In der System.conf
  - In der SmartMeta.conf
  - In der FTS.conf
  - In der Index.conf

13.4 Relevante Dateien

Die Umstrukturierungsaufgaben werden technisch auf Basis einer ToDo-Liste durhcgeführt. Die ToDo-Liste für die Umstrukturierungsaufgaben heisst `/site/logs/moving.todo`. Diese Datei bleibt auch nach einer erfolgreichen bearbeitung so lange bestehen, biss die nächste Umstrukturierungsaufgabe gestartet wird.

Während die Barbeiung läuft, wird die Oberfläche über Ajax-Requests regelmässg aktualisiert, der Fortschritt wird intern in der Datei `/site/logs/moving_ajax.log` `moving_process_ajax.output` `moving_create_ajax.output` festgehalten. Diese Datei wird zum Ende der Umstrukturierung automatisch gelöscht.

Das Protokoll über Umstrukturierung wird in die Datei `/site/logs/moving.log` geschrieben. z.B.:

```
[Do Sep 09 17:24:59 2010] [info] Start creating todo list.
[Do Sep 09 17:24:59 2010] [info] Scheduled document '/47/50/54/100' for moving.
[Do Sep 09 17:24:59 2010] [info] Scheduled document '/47/49/51/96' for moving.
[Do Sep 09 17:24:59 2010] [info] Scheduled document '/47/49/53/99' for moving.
[Do Sep 09 17:24:59 2010] [info] Scheduled document '/47/49/52/97' for moving.
[Do Sep 09 17:24:59 2010] [info] Scheduled grafiker file '/2/product_y' contained in document
[Do Sep 09 17:24:59 2010] [info] Scheduled grafiker file '/2/ii/product_a' contained in docume
[Do Sep 09 17:24:59 2010] [info] Scheduled grafiker file '/2/ii/product_c' contained in docume
[Do Sep 09 17:24:59 2010] [info] Scheduled grafiker file '/2/ii/product_g' contained in docume
[Do Sep 09 17:24:59 2010] [info] Finished creating todo list.
```

Die Zuordnung der akutell in der Oberfläche gestarteten Umstrukturierung zu der ToDo-Datei erfolgt über ein Ticket. Die ID des Tickets wird in der Bedienungsoberfläche von einem Schritt zum nächsten mitgegeben. Auf diesem Weg lassen sich eigene Prozesse eindeutig erkennen. Die zusätzlich benötigten Informationen werden in der Datei `/site/logs/moving.ticket` abgelegt. Diese Datei wird zum Ende der Umstrukturierung automatisch gelöscht.

Kapitel 14. Der Daemon Hermes

Bei einem Daemon handelt es sich um einen Prozess, der ununterbrochen im Hintergrund ausgeführt wird und nach einem bestimmten Zeitplan oder aufgrund bestimmter System-Ereignisse definierte Funktionen ausführt.

Der Daemon von **imperia** wird durch die Skripte `/site/bin/site_hermes_unix.pl` bzw. `/site/bin/site_hermes_win.pl` und `/site/bin/site_hermes.pl` repräsentiert und zeichnet sich durch einen modularen Aufbau aus. Alle Funktionen sind in externe Plug-Ins ausgelagert. Daraus ergibt sich ein Maximum an Flexibilität und Skalierbarkeit.

Er ist für alle automatischen Funktionen sowohl auf dem Produktionssystem als auch auf den Zielsystemen von **imperia** verantwortlich. Dazu gehören die Überwachung von Verzeichnissen, die Ausführung von Systemdiensten und SiteActives.

Für Testzwecke ist es sinnvoll, den Daemon bei Bedarf zu starten. Er bietet zusätzlich einen Debug-Modus zur Fehlersuche, die Sie durch eine entsprechende Kommandozeilenoption aktivieren können.

Die Ausführungseigenschaften der einzelnen Funktionen der Plug-Ins werden in einer Tabelle gespeichert, die darüber Auskunft gibt, wann und unter welchen Bedingungen ein Plug-In ausgeführt wird.

14.1 Starten des Hintergrund-Daemons Hermes

Es empfiehlt sich, den Daemon im Hintergrund auszuführen. Unter Windows®-Betriebssystemen ist die zusätzliche Registrierung des Daemons als Systemdienst empfehlenswert. Der Hermes-Daemon muss auf dem Entwicklungssystem und auf den Zielsystemen laufen.

14.1.1 Starten des Daemons unter UNIX®

Unter UNIX® sollte Hermes im Hintergrund beim Starten des Systems ausgeführt werden. Um den Hintergrund-Daemon manuell unter UNIX® zu starten, gehen Sie wie folgt vor:

1. Öffnen Sie eine Konsole und wechseln Sie in das Verzeichnis `site/bin` Ihrer **imperia** -Installation (Entwicklungs- bzw. Zielsystem).
2. Geben Sie dort folgenden Befehl ein:

```
nohup ./site_hermes_unix.pl &
```

14.1.2 Starten des Daemons unter Windows®

Um den Hintergrund-Daemon unter Windows® manuell zu starten, gehen Sie wie folgt vor:

1. Öffnen Sie eine DOS-Box und wechseln Sie in das Verzeichnis `site\bin` Ihrer **imperia** -Installation (Entwicklungs- bzw. Zielsystem). Geben Sie folgenden Befehl ein:

```
perl site_hermes_win.pl
```

2. Lassen Sie die DOS-Box geöffnet.

14.1.3 Der Daemon als Systemdienst unter Windows®

Wichtig:

Alle benötigten Dateien finden Sie im Verzeichnis `site/util/XYNTService` *der* **imperia** *Installation.*

14.1.3.1 Installation

Das Programm 'XYNTService' bietet die Möglichkeit, den Hermes als Systemdienst im Hintergrund unter Windows zu starten. Dafür ist die Anpassung der Datei 'XYNTService.ini' von Nöten, welche sich im Verzeichnis des Programms befinden muss. Anzupassen sind hier die Einträge 'CommandLine' und 'WorkingDir' um den Dienst an die Systemumgebung anzupassen.

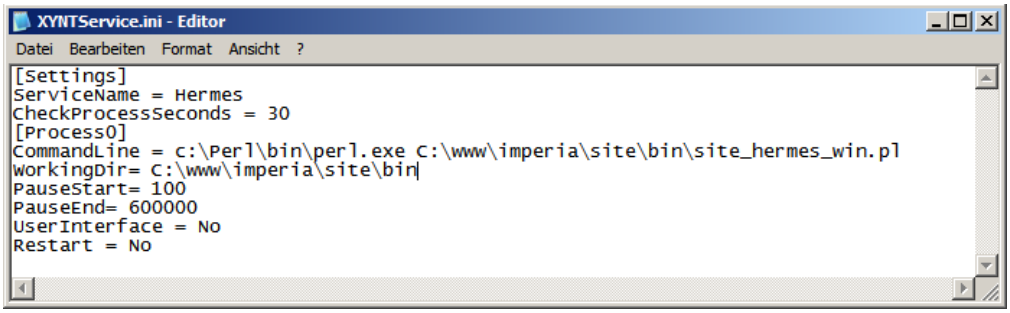


Abb. 14.1: Beispiel Konfiguration der XYNTService.ini

Weiterhin besteht die Möglichkeit, den Eintrag 'PauseEnd' nach Gebrauch zu verändern. 'PauseEnd' definiert die Zeit in Millisekunden, die beim Stop des Dienstes oder dem Herunterfahren des Systems dem Dienst (Hermes) verbleibt, um den Prozess ordentlich zu beenden. Im Beispiel werden 600000 Millisekunden gewählt, was 10 Minuten entspricht. Allerdings kann dieser Wert nicht ausreichend sein, falls das Hermes-Skript zu sehr ausgelastet ist. Verstreicht die Zeit, wird das Hermes-Skript vom Programm 'XYNTService.exe' umgehend beendet. Um den Systemdienst zu installieren, öffnen Sie eine DOS-Box, wechseln in das Verzeichnis, indem das Programm 'XYNTService.exe' liegt und geben folgenden Befehl ein:

```
XYNTService.exe -i
```

Wenn der Dienst erfolgreich installiert wurde, finden Sie im Windows Service Manager den Dienst 'Hermes' (der Name wird durch den Eintrag 'ServiceName' der 'XYNTService.ini' definiert) in der Liste der Dienste vor. Bei jedem Windows-Start wird der Dienst nun gestartet. Der Dienst kann auch manuell im Windows Service Manager oder durch folgenden Befehl einer DOS-Box gestartet werden:

```
net start Hermes
```

14.1.3.2 Deinstallation

Möchten Sie den Systemdienst entfernen, öffnen Sie eine DOS-Box, wechseln in das Verzeichnis, indem sich das Programm 'XYNTService.exe' befindet und geben folgenden Befehl ein:

```
XYNTService.exe -u
```

Wenn der Dienst erfolgreich destalliert wurde, ist der Eintrag 'Hermes' definiert durch den Eintrag 'ServiceName' der 'XYNTService.ini') in der Liste der Systemdienste im Windows Service Manager verschwunden.

14.2 Kommandozeilenoptionen

Über teilweise betriebssystemspezifische Kommandozeilenoptionen können Sie die Funktionsweise des Daemons beeinflussen.

14.2.1 Verwendung

```
site_hermes_unix.pl [OPTIONEN] | site_hermes_win.pl [OPTIONEN]
```

14.2.2 Optionen

Folgende Kommandozeilenoptionen sind, soweit nicht anders erwähnt, unabhängig vom Betriebssystem verfügbar:

Option *-b, --background*

Diese Option ist nur unter Unix® wirksam. Nach dem Start entkoppelt sich der Prozess vom Terminal und läuft im Hintergrund weiter.

Option *-d, --debug*

Diese Option aktiviert den Debug-Modus des Daemons. Es werden alle Aktivitäten des Daemons in der Log-Datei `hermes.log` im Verzeichnis `<ServerRoot>/site/logs` protokolliert.

Option *-e, --exit*

Verwenden Sie diese Option, damit der Daemon sich nach dem ersten Durchlauf automatisch beendet.

Option *-h, --help*

Diese Option zeigt einen kurzen Hilfetext in der Konsole/DOS-Box. Er enthält alle verfügbaren Kommandozeilenoptionen und eine kurze Beschreibung des Daemons. Dieser beendet sich sofort nach Anzeigen des Hilfetextes automatisch.

Option *-k restart*

Mit dieser nur unter Windows® verfügbaren Option veranlassen Sie einen Neustart des Hintergrund-Daemons. Das Skript entfernt vor dem Neustart automatisch die PID des laufenden Prozesses.

Option *-k stop | quit | exit*

Mit dieser nur unter Windows® verfügbaren Option beenden Sie den Hintergrund-Daemon. Das Skript entfernt dabei automatisch die PID des laufenden Prozesses.

Option *-p, --plug-in-debug*

Nutzen Sie diese Option, wenn Sie die Ausführung etwaiger Hermes-Plug-Ins debuggen möchten. Die Debug-Meldungen erscheinen in der Protokolldatei des Daemons `site/logs/hermes.log`.

Option *-s, -i, --seconds, --sleep=SECONDS*

Verwenden Sie diese Option, um eine Anzahl von Sekunden anzugeben, die der Daemon nach jedem Durchlauf pausieren soll.

Option *-t, --table*

Diese Option zeigt die Tabelle aller verfügbaren Plug-Ins. Der Daemon beendet sich sofort nach Anzeigen der Tabelle.

Option *-u USER, --user=USER*

Diese Option ist nur unter UNIX® wirksam. Wenn Sie den Daemon mit Root-Rechten starten, müssen Sie damit den User übergeben, unter dessen System-Konto der Daemon laufen soll. Aufgrund der UNIX®-Zugriffsrechte-Struktur muss der Daemon unter demselben User-Account laufen wie der Webserver. Dies ist häufig der User `nobody` oder auch `wwwrun`.

Option *-v, -z, --verbose*

Mit dieser Option aktivieren Sie die Ausgabe zusätzlicher Meldungen im Error-Log, die sonst nicht erscheinen.

14.2.3 Log-Datei

Alle Aktivitäten des Daemons werden in der Log-Datei `hermes.log` im Verzeichnis `site/logs` Ihrer **imperia** -Installation protokolliert. Abhängig vom eingestellten Debug-Modus können Menge und Art der protokollierten Informationen stark variieren. Hier ein typischer Ausschnitt aus einer Log-Datei:

```
[Wed Jun 10 16:07:17 2007] HERMES IS RUNNING ...
[Wed Jun 10 16:07:18 2007] Hermes running with pid 3516
[Wed Jun 10 16:07:18 2007] FOUND 8 PLUGINS (SiteActive, ArchiveMinimizer,
     Import, ExpiryDate, CleanUp, MessageCheck, LogRotate, AutoPublish)
[Wed Jun 10 16:07:18 2007] SCHEDULING PLUGINS
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN SiteActive
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN ArchiveMinimizer
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN Import
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN ExpiryDate
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN CleanUp
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN MessageCheck
[Wed Jun 10 16:07:19 2007] SUCCESSFULLY LOADED PLUGIN LogRotate
[Wed Jun 10 16:07:19 2007] SUCCESSFULLY LOADED PLUGIN AutoPublish
[Wed Jun 10 16:07:19 2007] STARTUP DONE

HERMES IS RUNNING ...
[Wed Jun 10 16:07:19 2007] [MessageCheck] Processing notification
                           '469df2e40360bf2.free'
[Wed Jun 10 16:07:19 2007] [MessageCheck] Processing notification
                           '469df2e403606e0.free'
[Wed Jun 10 16:07:19 2007] [MessageCheck] Processing notification
                           '469df2e40360f41.free'
[Wed Jun 10 16:07:19 2007] [Import] looking for configuration file
                           C:/imperia/dev/sanne/site/config/import.conf
[Wed Jun 10 16:07:19 2007] [Import] not found, falling back to hardcoded
                           defaults
[Wed Jun 10 16:07:19 2007] [Import] successfully (re)initialized plugin
                           Ascii
[Wed Jun 10 16:07:19 2007] [Import] successfully (re)initialized plugin
                           ImperiaXML
[Wed Jun 10 16:07:19 2007] [Import] successfully (re)initialized plugin
                           Meta
[Wed Jun 10 16:07:49 2007] SCHEDULING HERMES FOR QUIT ...
[Wed Jun 10 16:07:50 2007] EXIT SIGNAL RECEIVED.
[Wed Jun 10 16:07:50 2007] HERMES IS GOING AWAY...
[Wed Jun 10 16:07:50 2007] TERMINATING PID: 2912
```

Hinweis:

Obenstehender Dateiauszug ist aus Darstellungsgründen umgebrochen.

Die Prozess-ID des Daemons wird zusätzlich in der Datei `hermes.pid` im Verzeichnis `site/logs` gespeichert. In der Regel existiert diese Datei nur, wenn der Daemon läuft. Sie wird gelöscht, sobald der Daemon normal beendet wird.

14.2.4 Plug-In-Tabelle

Beim ersten Start des Daemons werden alle gefundenen Plug-Ins in die Ausführungstabelle aufgenommen. Jede Funktion eines Plug-Ins erhält eine ID, anhand derer sie eindeutig identifiziert werden kann.

Die Tabelle enthält neben dem Namen (NAME), der Funktion (FUNCTION) und der ID (ID) alle weiteren Eigenschaften des Plug-Ins. Hierzu gehören:

die Ausführungszeit (EXEC_TIME),
die maximale Laufzeit (MAX_RUNTIME),
die Priorität (PRIORITY),
der Zeitpunkt der letzten Ausführung (LAST_RUN).

Die Einträge in der Plug-In-Tabelle werden nach Ausführungspriorität aufgelistet. Die Spalte der Ausführungszeit (EXEC_TIME) lässt mehrere Eintragsformate zu:

Zeitintervall in Sekunden (z. B.: 300 => 5 Minuten, 0 => immer),
fester Zeitpunkt in hh:mm[:ss] (z. B.: '0:00' => Mitternacht),
keine Ausführung aufgrund der Kennung `no`.

Die Tabelle bleibt nach der initialen Erstellung auch bei Beendigung des Daemons gültig.

Plug-Ins werden nur beim Start des Hintergrund-Daemons geladen. Um ein Plug-In zu entfernen, muss der Hintergrund-Daemon angehalten werden. Danach kann man das entsprechende Plug-In aus dem Verzeichnis `site/modules/core/Dynamic/Hermes` entfernen und den Hintergrund-Daemon wieder starten.

14.3 Hermes-Import

Für den Import von Daten mit Hilfe des Hintergrund-Daemons Hermes sind im Lieferumfang von **imperia** mehrere Plug-Ins für verschiedene Formate enthalten. Der Hermes enthält eine fest programmierte Standard-Import-Konfiguration, mit der die Einstellungen für diese Standard-Formate festgelegt werden. Folgende Formate können importiert werden:

ASCII-Dateien mit der Endung `.simple`,
XML-Dateien, die nach dem Schema `red*.xml`benannt sind,
imperia -Metadateien mit der Endung `.meta`.

Um andere Formate zu importieren, muss ein entsprechendes Import-Plug-In programmiert und die Standard-Import-Konfiguration erweitert werden. Die geänderte Konfiguration muss in der Datei `/site/config/import.conf` gespeichert werden.

Diese Datei existiert nach der Installation nicht, sondern muss angelegt werden, da die Standard-Import-Konfiguration fester Bestandteil des Moduls `/site/modules/core/Dynamic/Hermes/Import.pm` ist. Die in der Datei `import.conf` eingestellte Konfiguration überschreibt die Standard-Import-Konfiguration. Soll neben der neuen Konfiguration die Standard-Import-Konfiguration erhalten bleiben, muss diese aus dem Modul in die `import.conf` kopiert werden.

Alle importierten Dokumente enthalten nach dem Import eine zusätzliche Meta-Variable `__imperia_imported`, dessen Wert ein Zeitstempel ist, der die Anzahl der Sekunden seit einem bestimmten Zeitpunkt (Epoche) enthält. Dieser Zeitpunkt ist auf den meisten Systemen der 1. Januar 1970, 00:00 GMT.

14.3.1 Standard-Import-Konfiguration des Hermes-Daemons

Die Standard-Import-Konfiguration enthält Abschnitte für alle im Lieferumfang von **imperia** enthaltenen Import-Plug-Ins. Jedes Format wird in der Konfigurationsdatei in einem `filetype`-Tag definiert. Dieses Tag muss folgende Tags enthalten:

- Ein `mask`-Tag, das einen Regulären Ausdruck enthält, der mit dem Namen der zu importierenden Dateien verglichen wird.
- Ein `plugin`-Tag, das den Namen des zu verwendenden Plug-Ins enthält.
- Ein optionales `data`-Tag, mit dem andere Daten an das Plug-In übergeben werden können.

Achtung:

Es ist nur ein `data`*-Tag je* `filetype`*-Tag erlaubt. Weitere* `data`*-Tags werden ignoriert.*

Die Standard-Import-Konfiguration ist wie folgt:

```
<?xml version="1.0" encoding="us-ascii"?>
<import_config>

  <filetype>
    <mask>^.*\.simple$</mask>
    <plugin>Ascii </plugin>
    <data>zusätzliche Daten für das ASCII-Plug-In</data>
  </filetype>
```

```
  <filetype>
    <mask>.*/red[^/]*\.xml$</mask>
    <plugin>ImperiaXML </plugin>
    <data>zusätzliche Daten für das ImperiaXML-Plug-In</data>
  </filetype>

  <filetype>
    <mask>^.*\.meta$</mask>
    <plugin>Meta</plugin>
    <data>zusätzliche Daten für das Meta-Plug-In</data>
  </filetype>
</import_config>
```

Mit dieser Import-Konfiguration können folgende Formate importiert werden:

- ASCII-Dateien, die auf `.simple` enden und durch das ASCII-Plug-In verarbeitet werden.
- XML-Dateien, die auf den Regulären Ausdruck `.*/red[^/]*\.xml$` passen und durch das Imperia-XML-Plug-In verarbeitet werden.
- **imperia** -Meta-Dateien, die auf `.meta` enden und durch das Meta-Plug-In verarbeitet werden.

Sollen andere Formate importiert werden, muss die Konfiguration angepasst werden. Für jedes weitere Format muss ein entsprechendes Plug-In vorhanden sein.

Zur Programmierung eigener Plug-Ins empfiehlt sich die Datei `/site/modules/core/Dynamic/Import/Ascii.pm` als Einstiegshilfe.

14.3.2 Zielrubrik bestimmen

Ein häufiger Grund für das Fehlschlagen des Imports ist das Fehlen der Rubrikinformationen, so dass **imperia** nicht entscheiden kann, in welche Rubrik ein Dokument importiert werden soll.

Plug-Ins entscheiden dies aufgrund des Inhalts des `data`-Tags in der Konfigurationsdatei, wobei versucht wird, die dort gefundenen Informationen zuerst der Meta-Variablen `__imperia_category` (Name der Rubrik) und, wenn dies fehlschlägt, der Meta-Variablen `__imperia_node_id` (Ziffernfolge) zuzuordnen.

Enthält die Konfigurationsdatei keine Informationen über die zu verwendende Rubrik, prüft **imperia** , ob in den zu importierenden Dateien jeweils eine Rubrik angegeben ist. Ist dies nicht der Fall, schlägt der Import fehl und Hermes liefert eine Fehlermeldung.

Existieren im Rubrikenbaum mehrere Rubriken mit dem gleichen Namen, verwendet Hermes die erste passende Rubrik. Um eine bestimmte Rubrik zu verwenden, muss diese über die Node-ID eindeutig identifiziert werden.

Eine NodeID besteht aus zwei Teilen:

- der Rubrik-ID, zum Beispiel `/12/24/35/`
- der Dokument-ID, zum Beispiel `130336`.

Die vollständige NodeID aus diesen beiden Teilen stellt sich wie folgt dar: `/12/24/35/130336`. Sowohl die Dokument-ID als auch jeder einzelne Abschnitt der Rubrik-ID sind garantiert eindeutig, da **imperia** sie aus einem automatisch inkrementierten Zähler generiert.

Enthält die Meta-Variable `__imperia_node_id` eine gültige NodeID, wird die Rubrik aus dieser Information gebildet, wobei die Dokument-ID nicht beachtet wird. Die Dateien werden dann in die gefundene Rubrik importiert.

14.3.3 Vorhandene Dokumente beim Import mergen

Neben der Möglichkeit, Dokumente anhand der NodeID in die richtige Zielrubrik zu importieren, gibt es zwei weitere Meta-Variablen, die dazu verwendet werden können, vorhandene und importierte Dokumente zu "mergen":

- `__imperia_clobber_by_name = 1 | 0`
- `__imperia_clobber_by_id = nodeID, z. B. /1/2/3`

Wenn Sie die Meta-Variable `__imperia_clobber_by_name` aktivieren, indem Sie sie auf 1 setzen, wird im Dokumenten-Baum nach einem Dokument gesucht, dass die gleiche URL wie das zu importierende Dokument hat. Wird ein solches Dokument gefunden, werden die beiden zusammengeführt, d. h. alle zusätzlichen Informationen aus der XML-Datei werden als Metafelder in Meta::Info des bereits im Archiv existierenden Dokumentes geschrieben.

Die Meta-Variable `__imperia_clobber_by_id` hat die gleiche Funktion, hierbei wird jedoch nicht nach einer URL, sondern nach einer NodeID gesucht. Diese Alternative ist schneller als die Suche nach der URL.

In beiden Fällen wird nur die ausgewählte Rubrik (siehe Kapitel 14.3.2) durchsucht.

14.3.4 Aufbau der Import-Daten

In diesem Abschnitt wird das jeweilige Format der Dateien beschrieben, die mit den Standard-Import-Plug-Ins importiert werden können.

14.3.4.1 ASCII-Import

Eine Datei, die durch das ASCII-Import-Plug-In importiert werden soll, muss aus Schlüssel-Wert-Paaren bestehen, wobei Schlüssel und Wert durch Gleichheitszeichen voneinander getrennt werden. Beispiel:

```
filename = index.html
directory = /import/ascii
title = Import-Test
metapage = movies
author = superuser
docid = 131875
__imperia_node_id = /4/10
exoiry_date = 2002-03-11 10:06
```

14.3.4.2 XML-Import

Eine Datei, die mit Hilfe des Imperia-XML-Plug-Ins importiert werden soll, muss für jedes Metafeld ein entsprechendes Tag beinhalten, das wiederum den Wert des Metafeldes enthalten muss.

Beispiel:

```
<?xml version="1.0" encoding="ASCII" standalone="yes"?>
<IMPERIA_CONTENT>
  <filename>index.html</filename>
  <directory>/import/xml</directory>
  <title>Import-Test</title>
  <metapage>movies</metapage>
  <author>superuser</author>
  <docid>131876</docid>
  <__imperia_node_id>/4/11</__imperia_node_id>
  <expiry_date>2002-03-11 10:06</expiry_date>
</IMPERIA_CONTENT>
```

14.3.4.3 Meta-Import

Bei **imperia** -Metadateien handelt es sich um Binärdateien, die nicht ohne Weiteres von Hand erzeugt werden können. Dieser Import wird dazu verwendet, bereits bestehende **imperia** -Dokumente von einer **imperia** -Installation in eine andere (auf Basis der gleichen Programm-Version) zu übertragen.

Kapitel 15. Navigationserzeugung

Basiert auf virtual includes im Webserver.

15.1 Konfiguration

Die Navigationserzeugung wird durch ein Konfigurationsdatei gesteuert. Als Grundlage kann das Beispiel `config/navigation_generator.conf.sample`verwendet werden. Diese Datei sollte dann in z.B. `config/navigation_generator.conf` umbenannt werden.

In der Konfiguration muss unter dem Hauptknoten <root> der allgemeine Konfigurationspunkt <common> vorhanden sein. Über diesen Punkt wird gesteuert, wie die zu erzeugenden Dateien heißen und wie sie auf das Livesystem transferiert werden.

Der Name der zu erzeugenden Dateien wird über das Feld

```
filename = "/ssi/nav/intra/<!--nav_element-->/<!--node_id-->/<!--lingua-->.ssi"
```

gesteuert. Dabei stehen die Werte <!--directory-->, <!--node_id-->, <!--cat_id-->, <!--lingua-->, <!--nav_element--> zur Verfügung.

Bei der Erstellung der Navigationselemente wird zusätzlich eine XML-Datei für den Importer geschrieben, in der alle neu erzeugten Dateien in Form eines IBlocks aufgelistet sind.

Über die transfer_cat_id wird die Node-Id einer Rubrik angegeben, hinter der ein nicht-interaktiver Workflow liegt. Das Template in dieser Rubrik besteht nur aus dem Eintrag

```
<!--IMPERIABLOCK-->
<!--
<IMG SRC="<!--XX-IBLOCK-file-->">
-->
<!--/IMPERIABLOCK-->
```

Dabei ist wichtig und zu beachten, das der auskommentierte Bereich dafür sorgt, dass die im IBlock angegebenen Dateien nicht angezeigt, aber transferiert werden.

Neben dem allgemeinen Block (common) können beliebig viele Navigationselemente erzeugt werden. Üblicherweise eine Horizontalnavigation (hier h_nav), eine Vertikalnavigation (hier v_nav) und eine Sitemap (hier sitemap). Die Namen der Navigationselemente sind frei, wenn sie die Namenskonvention von XML nicht verletzen.

Am Beispiel der Horizontalnavigation soll die Funktionsweise kurz beschrieben werden.

```
<h_nav>
 file_startlevel = "0"
 file_maxlevel = "5"
 content_startlevel = "0"
 content_maxlevel = "7"

 li_snippet <<EOF
    #IF ("<!--XX-current-->")
        <strong><!--XX-name--></strong>
    #ELSIF ("<!--XX-active-->")
        <a href="<!--XX-link-->" title="<!--XX-description-->">
                <!--XX-name--></a> - <!--XX-ul_content-->
    #ENDIF
EOF

 ul_snippet <<EOF
  <!--XX-li_content-->
EOF
```

```
</h_nav>

    (Text zur Darstellung umgebrochen!)
```

Über `file_startlevel` und `file_maxlevel` wird gesteuert für welche Ebenen des Rubrikenbaumes Dateien erzeugt werden sollen. In diesem Fall werden nur Dateien bis zur 5. Ebene erzeugt.

Die Variablen `content_startlevel` und `content_maxlevel` steuern welche Ebenen des Rubrikenbaumes innerhalb der Dateien dargestellt werden sollen.

Wie jeder einzelne Rubrikeneintrag gestaltet wird, hängt nur von den Variablen `ul_snippet` und `li_snippet` ab. D.h. zur Darstellung des Baumes werden diese Variablen für jedes Element erzeugt und durch die Verwendung der Variablen <!--XX-li_content--> und <!--XX-ul_content--> ineinander Verschachtelt.

Zur Erstellung stehen folgende Variablen zur Verfügung:

- Name der Rubrik (sprachabhängig)

  <!--XX-name-->

- Beschreibung der Rubrik (sprachabhängig)

  <!--XX-description-->

- Das Verzeichnis

  <!--XX-directory-->

- Der Dateiname

  <!--XX-filename-->

- Ein Spezieller Link (__ip_link)

  <!--XX-link-->

- Das Linkziel

  <!--XX-linktarget-->

- Die Bearbeitungsebene

  <!--XX-level-->

- Der Index der Bearbeitungsebene

  <!--XX-levelindex-->

- Der ausgewähle Menüpunkt (Bool)

  <!--XX-current-->

- Aktive Menüpunkte (Bool)

  <!--XX-active-->

(siehe auch Abschnitt 15.2 **Erstellung der Navigations-Dateien** auf Seite 174)

15.2 Erstellung der Navigations-Dateien

Die Navigationserzeugung wird über einen Menüpunkt in einem administrativen Menü angestoßen. Über diesen Menüpunkt können alle Navigationselemente erzeugt werden.

Zu diesem Zweck wird in der Menükonfiguration bei „Link“ z.B.

```
NEWWIN:/cgi-bin/site_navgen.pl?
    conf=/config/navigation_generator.conf&
    rootNodeId=/13
```

eingetragen. (Text wurde hier umgebrochen!)

Dieses Skript erzeugt die Codeschnipsel, die in den HTML-Seiten eingebunden werden. Für jede Navigationsebene wird ein Codeschnipsel erzeugt. Die Namen der erstellten SSI-Dateien werden über die Konfigurationsdatei festgelegt.

`conf=/config/navigation_generator.conf` beschreibt die Datei die zur Erstellung der Navigationselemente herangezogen werden soll. `rootNodeId=/13` beschreibt die Node-Id bei der die Navigationserzeugung starten soll.

(siehe auch Abschnitt 15.1 **Konfiguration** auf Seite 173)

Die Navigationserzeugung wird von einem berechtigten Imperia-Benutzer ausgeführt. Hierbei wird die komplette Navigation zunächst auf dem Basissystem erzeugt.

Nach der Erzeugung der Navigation existieren die Include-Dateien zunächst nur auf dem Basissystem. Zur Übertragung auf den Live-Server wird ein Dokument von Imperia importiert. Die hierfür notwendigen Daten werden vom Skript erstellt und in Form eines XML-Files in das Import-Verzeichnis geschrieben (/site/import). Das XML-Dokument wird als neues Dokument in Imperia in eine hierfür bereitgestellten Rubrik importiert. Über den Workflow dieser Rubrik kann u.a. gesteuert werden, ob die Navigationsdateien direkt auf den Live-Server freigeschaltet (autopublish) oder ob die Navigationsdateien erst nach einem Genehmigungsschritt (approval) freigeschaltet werden.

Kapitel 16. Die imperia -Volltextsuche

16.1 Einführung

Die Suche ist integraler Bestandteil von **imperia** und dient dem Auffinden beliebiger Wörter oder Wortkombinationen in Dokumenten eines Projekts und der strukturierten Ausgabe der Suchergebnisse. Sie dient dazu, den Benutzern eines Zielsystems eine Möglichkeit zum Durchsuchen der dort vorhandenen Dokumente zu bieten. Sie können damit auch die Dokumente auf dem Produktionssystem durchsuchen. Dies ist sinnvoll, wenn die Möglichkeiten der in die **imperia** -Benutzeroberfläche integrierten Suche nicht ausreichen. Hierbei werden nur Dokumente indiziert, die den Workflow bereits verlassen haben.

Die Volltextsuche verarbeitet per Default ASCII-Dokumente (HTML, ASP ...). Mit zusätzlichen Hilfsprogrammen sind auch Microsoft-Office- Dokumente (*.doc, *.xls) und PDF-Dokumente durchsuchbar. Durch Gruppierungen kann die Suche auf bestimmte Bereiche eingeschränkt werden. Gruppierungen können auf Verzeichnis- oder auf Meta-Ebene erfolgen. Die Suche kann durch Verwendung unterschiedlicher Such- und Ergebnismasken für verschiedene Benutzergruppen angepasst werden. In den Ergebnislisten können konfigurationsabhängig beliebige Metainformationen angezeigt werden. Domains ermöglichen Ihnen die Verwaltung mehrerer Suchindizes. Auf diese Weise lassen sich beispielsweise die Datenbestände auf Systemen mit mehreren Mandanten sauber voneinander getrennt indizieren und durchsuchen.

Grundsätzlich indiziert die Volltextsuche den sichtbaren HTML-Text eines Dokuments. Es ist aber auch möglich, gezielt nicht sichtbare Metafeld-Inhalte zu indizieren und sichtbaren HTML-Code von der Indizierung auszuschließen.

16.2 Eingabe von Suchanfragen und Syntax für Suchbegriffe

Die Eingabe von Suchanfragen erfolgt über eigene Templates. Hinweise zur speziellen Syntax für die Erstellung von Suchtemplates finden Sie im Programmierhandbuch im Abschnitt "**Templates für die imperia -Volltextsuche**". Auch für die Eingabe von Suchbegriffen gibt es einige Syntaxregeln, mit deren Hilfe sich Anfragen gezielter durchführen lassen. Ein Suchbgeriff besteht aus einem oder mehreren Wörtern. Eine Verfeinerung der Suche und damit präzisere Suchergebnisse erzielen Sie durch die Verknüpfung mehrerer Suchwörter. Hierfür stehen Ihnen eine Gruppierungsfunktion, logische Operatoren und die Möglichkeit, gezielt nach Phrasen zu suchen, zur Verfügung. Außerdem können Sie bereits im Suchbegriff Ihre Anfrage nach dem Inhalt bestimmter Metafelder filtern. Die Syntax dieser speziellen Suchvarianten ist in den folgenden Abschnitten beschrieben. Der Abstand zwischen einzelnen Wörtern aus einem Suchbegriff im Text der gefundenen Dokumente spielt bei der Berechnung der Relevanz keine Rolle.

16.2.1 Groß- und Kleinschreibung

Die **imperia** -Volltextsuche ignoriert die Groß- und Kleinschreibung von Suchbegriffen.

16.2.2 Sonderzeichen und Interpunktion

Die Volltextsuche ignoriert Sonderzeichen und Interpunktion generell, da diese generell nicht mit indiziert werden. Aus diesem Grund gibt es auch keine Ausnahmen für feststehende Begriffe mit Sonderzeichen wie z. B. 'C++'. Es gibt allerdings bei der Verwendung von Sonderzeichen in Suchbegriffen einige Ausnahmen. Diese sind '"', '(', ')', ':', '*' und '|'. Mit diesen Sonderzeichen führen Sie die oben angeführten speziellen Suchvarianten aus. Einzelheiten hierzu entnehmen Sie bitte den folgenden Abschnitten.

16.2.3 Wildcards

Verwenden Sie den Stern '*' in Suchbegriffen, um damit eine Wildcard für einen oder mehrere Buchstaben einzusetzen. Je mehr Wildcards eine Suchanfrage enthält, desto ressourcenintensiver ist deren Verarbeitung. Daher ist die Volltextsuche per Default so konfiguriert, dass pro Suchbegriff maximal eine Wildcard möglich ist. Bei dieser Einstellung dürfen Suchbegriffe mit Wildcard aus Performancegründen nur aus einem Wort bestehen. Beispielsweise liefert eine Suche nach 'a*'' potenziell hunderte von Treffern. Diese mit weiteren per AND bzw. OR verknüpften Suchwörtern abzugleichen ist sehr ressourcenintensiv.

Verwenden Sie Wildcards bei der Default-Einstellung dennoch in Verbindung mit anderen Suchwörtern, erhalten Sie nicht die gewünschten Treffer. Die Volltextsuche ignoriert in diesem Fall die Wildcard. Eine Suche nach 'a*e AND Begriff' liefert statt Dokumenten mit Wörtern, die mit a beginnen und auf e enden und Begriff enthalten, Dokumente mit a oder (Default-Operator) e und Begriff. Wildcards, die nicht Bestandteil eines Suchworts sind (Leerzeichen davor und danach) ignoriert die Volltextsuche. Es ist also nicht möglich, eines oder mehrere Wörter in Suchphrasen (siehe Abschnitt 16.2.8 **Phrasensuche** auf Seite 177) durch eine Wildcard zu ersetzen.

Wenn Sie die Performance-Einbußen in Kauf nehmen und mehrere Wildcards pro Suchbegriff und einzelnem Suchwort sowie die Kombination von Suchwörtern mit und ohne Wildcards wünschen, können Sie dies in der Suchkonfiguration einstellen.

16.2.4 Alternative Schreibweisen / Worttrennungen

Eine Funktion zur Auffindung alternativer Schreibweisen für Komposita mit und ohne Bindestrich, getrennt- und zusammengeschrieben ist derzeit nicht implementiert. Getrennt geschriebene Wörter werden auch getrennt indiziert. Es ist nicht möglich, gezielt nach Wörtern in Bindestrich-Schreibweise zu suchen.

16.2.5 Umlaute

Die **imperia** -Volltextsuche unterscheidet bei Umlauten nicht zwischen den alternativen Schreibweisen. Die Suchbegriffe "Bürgermeister" und "Buergermeister" liefern also identische Ergebnisse.

16.2.6 Phonetisch ähnliche Alternativschreibweisen

Die Volltextsuche ignoriert phonetische Ähnlichkeit. Das gilt auch für Begriffe wie "Graphik" und "Grafik" etc.

16.2.7 Stopwörter

Die **imperia** -Volltextsuche bietet die Möglichkeit, eine Liste von Stopwörtern zu definieren. Die Wörter aus dieser Liste erscheinen nicht im Index. Die Volltextsuche ignoriert Stopwörter außerdem in Suchanfragen. Eine Anfrage, die ein Wort aus der Stopwortliste sowie weitere Suchwörter enthält, liefert also dasselbe Ergebnis wie eine Anfrage ohne das Stopwort. Dabei ist unerheblich, ob die einzelnen Suchwörter mit **UND** oder **ODER** verknüpft sind oder ob es sich um eine Phrasensuche handelt. Besteht ein Suchbegriff ausschließlich aus Stopwörtern, liefert er gar keine Treffer.

Im Ergebnis-Template haben Sie die Möglichkeit, gesonderte Rückmeldung bei Verwendung etwaiger Stopwörter in Suchanfragen zu geben.

Stopwortlisten sind global oder domainspezifisch konfigurierbar. Kombinationen aus beiden Varianten sind ebenfalls möglich. Einrichtung und Verwaltung von Stopwortlisten sind in Abschnitt 16.5.3.7 **Stopwortlisten anlegen und verwalten** auf Seite 213 beschrieben.

16.2.8 Phrasensuche

Schließen Sie einen oder mehrere Begriffe in doppelte Anführungszeichen ein, um diese als Suchphrase zu kennzeichnen. Die Suche liefert daraufhin nur Treffer, die alle Wörter der Phrase in exakt der selben Reihenfolge enthalten. Die Interpunktion im Originaldokument ist dabei irrelevant.

16.2.9 Kategorisierte Suchanfragen

Die Einschränkung der Suchanfrage auf bestimmte Kategorien erreichen Sie durch die Definition von MetaGroups bei der Indexerstellung, die Sie dann im Suchformular zur Auswahl stellen. Einzelheiten hierzu finden Sie in Abschnitt 16.5.3.3.2 **Tags in Domain-Blöcken** auf Seite 187.

16.2.10 URL-Suche

Eine gezielte URL-Suche nach dem Muster der Google-Suchmaschine können Sie auch mit der **imperia** -Volltextsuche umsetzen. Definieren Sie hierzu in den Dokumenten ein Metafeld namens site. Suchanfragen mit der Syntax 'site:www.myurl.de' liefern dann ein entsprechendes Ergebnis. Syntaktisch handelt es sich dabei um einen Filter nach Metafeldinhalten. Lesen Sie hierzu auch Abschnitt 16.2.13 **Nach Metafeldinhalten filtern** auf Seite 178.

16.2.11 Wortstammsuche

Momentan ist die Suche nach Alternativ-Wörtern mit dem gleichen Wortstamm nicht implementiert.

16.2.12 Gruppierung

Durch Einklammern "()" können Sie eines oder mehrere Wörter in Suchbegriffen gruppieren. Achten Sie dabei auf korrekte Klammersetzung. Bei fehlenden öffnenden oder schließenden Klammern funktioniert die Gruppierung nicht. Stattdessen interpretiert die Volltextsuche die einzelnen Klammern als Bestandteile des Suchbegriffs.

16.2.13 Nach Metafeldinhalten filtern

Um Suchanfragen nach bestimmten Metafeldinhalten zu filtern, notieren Sie bei der Eingabe des Suchbegriffs Metafeldname:Begriff bzw. Metafeldname:|Begriff Begriff2|. Das zweite Beispiel mit den Pipe-Zeichen beinhaltet eine implizite UND-Verknüpfung der beiden Suchbegriffe. Es erscheinen nur Suchergebnisse, auf die der definierte Filter zutrifft, in denen es also das betreffende Metafeld mit dem passenden Inhalt gibt.

16.2.14 Suchwörter verknüpfen und ausschließen, logische Operatoren

Mit den Operatoren AND und OR verknüpfen Sie Teile des Suchbegriffs miteinander. Die Verknüpfung 'Wort1 AND Wort2' liefert Ihnen nur Treffer, in denen beide Suchwörter enthalten sind. Bei 'wort1 OR Wort2' erhalten Sie zudem Treffer, in denen eines der beiden Suchwörter enthalten ist. Wollen Sie bestimmte Wörter aus der Suche ausschließen, nehmen Sie diese mit einem vorangestellten NOT in den Suchbegriff auf. Die Groß- und Kleinschreibung ist bei den drei Operatoren AND, OR und NOT irrelevant. In der Suchkonfiguration (Datei `site/config/fts.conf`) können Sie sprachspezifische Synonyme für die logischen Operatoren angeben. Außerdem funktionieren folgende Sonderzeichen als "Shortcuts" für logische Operatoren:

Operator	Shortcut
AND	+
OR	?
NOT	-, !

Tabelle 16.1. Shortcuts für logische Operatoren

Logische Operatoren lassen sich beliebig mit Gruppierungen kombinieren. Mehrere Suchwörter ohne Operator im Suchbegriff verknüpft die Volltextsuche per Default mit OR. Diesen Default können Sie in der Suchkonfiguration ändern (siehe Abschnitt 16.5.3.4.3 **Direktiven in Option-Blöcken** auf Seite 204, Direktive `defaultOperator`).

Eine Besonderheit stellen zusammengesetzte Suchworte mit Bindestrich dar. So wird bei der Suche nach "Handball-WM" intern folgendes gesucht: (handball OR wm) OR "handball-wm" OR "handballwm". Ist als Standardoperator "AND" gesetzt, so wird folgendes gesucht: (handball AND wm) OR "handball-wm" OR "handballwm".

16.3 Anzeige von Suchergebnissen

Auch die Darstellung von Suchergebnissen erfolgt über eigene Templates. Die spezielle Syntax für die Erstellung von Ergebnistemplates ist im Programmierhandbuch im Abschnitt "**Templates für die imperia -Volltextsuche**" erklärt. Die Sortierung der Trefferliste erfolgt per Default absteigend nach Relevanz. Weitere Sortieroptionen lassen sich durch die Konfiguration von Ergebnisfiltern aktivieren. Lesen Sie hierzu Abschnitt 16.3.3 **Ergebnisfilter** auf Seite 179, bzw. Abschnitt 16.5.3.4.6 **UserSort: Sortierung der Treffer bestimmen** auf Seite 210 und die darin enthaltenen Erläuterungen zur Direktive `sortBy`. Bei der Anzeige der einzelnen Treffer haben Sie die Möglichkeit, eine frei konfigurierbare Menge Kontext anzuzeigen, in dem der gefundene Suchbegriff hervorgehoben erscheint.

16.3.1 Kontextanzeige

Bei der Anzeige von Treffern gibt die Volltextsuche für jeden Treffer einen Textausschnitt mit dem Suchbegriff aus. Die Wortanzahl für diesen Textausschnitt können Sie bestimmen, nicht jedoch die Position des Suchbegriffs in dem Ausschnitt. Bei aus mehreren Wörtern zusammengesetzten Suchbegriffen erscheint der erste Textabschnitt aus einem gefundenen Dokument, in dem die meisten der gesuchten Wörter vorkommen und der die vorgegebene Wortanzahl nicht überschreitet. Selbst wenn das Dokument alle gesuchten Wörter enthält, müssen diese nicht unbedingt alle in dem dargestellten Kontext erscheinen, wenn die eingestellte Wortanzahl dafür nicht ausreicht. Auch muss bei einer Phrasensuche nicht zwangsläufig der erste Abschnitt im Dokument der sein, der die gesuchte Phrase enthält. Sollten im selben Dokument vorher alle gesuchten Wörter in einem darstellbaren Textabschnitt in anderer Reihenfolge vorkommen, zeigt die Volltextsuche diesen an.

16.3.2 Highlighting

Bei der Ausgabe der Treffer lassen sich Suchbegriffe hervorgehoben darstellen. Die hierfür notwendigen Anweisungen tragen Sie im Ergebnistemplate ein. Für die Hervorhebung in Suchbegriffen mit Wildcards stehen zudem spezielle Hervorhebungsmodi und CSS-Klassen zur Verfügung. Lesen Sie hierzu den Abschnitt "**Templates für die imperia -Volltextsuche**" im Programmierhandbuch. **imperia** enthält eine Standard-CSS-Datei für die Darstellung von Hervorhebungen. Durch die Definition einer lokalen CSS-Klasse im Ergebnistemplate können Sie dieses Standard-Design der Hervorhebung durch eigene Formatierungen ersetzen.

16.3.3 Ergebnisfilter

Normalerweise schickt **imperia** direkt nach der Verarbeitung der Suchanfrage die Ergebnisliste an den anfragenden Client zurück. Eine offene Plug-In-Schnittstelle ermöglicht es, die Trefferliste vor der Darstellung aufzubereiten. So lassen sich beispielsweise spezielle Sortiermethoden oder auch Manipulationen an der Darstellung der Metainformationen der gefundenen Dokumente implementieren.

Ergebnisfilter-Plug-Ins legen Sie im Verzeichnis `site/modules/core/Dynamic/FTSResultFilter` Ihres Systems ab. Dort gibt es auch schon einige Beispiele. Die verfügbaren Methoden für die Aufbereitung der Trefferliste sind im POD des Moduls `site/modules/core/Imperia/FTS/ResultFilterPlugin.pm` beschrieben.

Für Hinweise zur Aktivierung von Ergebnisfiltern in der Volltextsuche lesen Sie bitte Abschnitt 16.5.3.4.3, „resultFilterPlugin“.

16.4 Berechnung der Treffer-Relevanz und Boni

Die Berechnung der Relevanz eines Dokuments in einer Trefferliste erfolgt in mehreren Teilschritten. Bei der Indizierung ermittelt die Volltextsuche für jedes im Text enthaltene Wort einen Relevanzwert. Dieser ergibt sich daraus, in wie vielen Dokumenten das betreffende Wort jeweils wie häufig vorkommt. Die endgültige Relevanz der einzelnen Wörter im Index ergibt sich dann aus der Relation zum Durchschnittswert. Wenig aussagekräftige Wörter wie "und", "ist" etc. erhalten so eine vergleichsweise geringe Relevanz, da sie in sehr vielen Dokumenten sehr häufig vorkommen.

Bei einer Suchanfrage ermittelt die Volltextsuche die Relevanz der gefundenen Dokumente aus dem Produkt der Relevanz des Suchbegriffs und seiner Häufigkeit im jeweiligen Dokument. Besteht der Suchbegriff aus mehreren Wörtern, errechnet sich die Relevanz der Dokumente aus der Summe der Werte der einzelnen Wörter. Die Trefferrelevanz der Dokumente können Sie durch die Vergabe von Boni in zweierlei Hinsicht beeinflussen: Boni lassen sich anhand der Position eines Wortes im Dokument und datumsbezogen definieren. Lesen Sie hierzu Abschnitt 16.5.3.3.1, „<bonus>“ und Abschnitt 16.5.3.4.3 **Direktiven in Option-Blöcken** auf Seite 204, Direktiven zu `bonus_date_`.

Auch wenn der letztendlich ermittelte Relevanzwert eine absolute Zahl ist, erfolgt die Ausgabe als Prozentangabe. Dabei dient das Dokument mit dem höchsten Wert als Ausgangspunkt für die Umrechnung.

16.5 Einrichten der imperia -Volltextsuche

Die Volltextsuche verwendet zwei Skripte: `site/bin/fts_index.pl` für die Indexerstellung und `cgi-bin/fts_search.pl` für die Verarbeitung der Suchanfragen.

Außerdem gibt es das Hilfsskript `fts_conf_convert.pl` zum Konvertieren einer bestehenden Konfiguration älterer Versionen der **imperia** -Volltextsuche in das neue Format. Die gesamte Konfiguration wird über die Konfigurationsdateien `/site/config/index.conf` und `/site/config/fts.conf` gesteuert. Mit `/site/config/index.conf.sample` existiert eine Beispielkonfiguration, in welcher die Funktion jedes Parameters kurz erläutert wird. Auch solche, auf welche in dieser Dokumentation nicht explizit eingegangen wird.

Eine Anleitung zum schnellen Start finden Sie im Abschnitt 16.5.2 **Schnellstartanleitung** auf Seite 180. Auf die einzelnen Skripte wird in späteren Abschnitten detailliert eingegangen.

16.5.1 Voraussetzungen

Die **imperia** -Volltextsuche benötigt folgende Perlmodule:

- BerkeleyDB >= 4.0. (empfohlene Version, mindestens jedoch Version 3.0)
- BerkeleyDB::Btree
- Config::General (*)
- File::Spec (*)
- File::Copy (*)
- File::Path (ab Perl 5.8.X Bestandteil des Interpreters)
- Tree::RedBlack (*)

Die mit einem Stern gekennzeichneten Module sind in der **imperia** -Distribution enthalten, die übrigen müssen Sie vor dem Einsatz der Volltextsuche selbst installieren. Diese Perl-Module können Sie kostenlos von *www.cpan.org* beziehen. Sollte eines der benötigten Module fehlen, bekommen Sie dies durch eine entsprechende Fehlermeldung angezeigt.

Ein optionales Modul ist folgendes:

- Compress::Zlib

Dieses Modul brauchen Sie nur, wenn Sie den Cache des Index aus Platzersparnisgründen komprimieren möchten.

Wenn Sie neben HTML- und einfachen Textdateien andere Dateitypen indizieren und durchsuchen wollen, z. B. PDF-Dokumente, benötigen Sie zusätzliche Parser-Programme, die Sie nachträglich installieren müssen.

Gegenwärtig gibt es folgende Parser-Plug-Ins :

Für Microsoft Office Dokumente das Programm Catdoc, zu beziehen über *http://www.45.free.net/~vitus/software/catdoc/* (Stand der Quellenangabe: April 2007). Benötigt wird die Version 0.94x, die unter der GPL 2 Lizenz steht. Es handelt sich dabei um eine Suite von Parsern, zu der folgende Programme gehören:

- DOC -Parser für Word-Dokumente
- XLS -Parser für Excel-Dokumente
- PPT -Parser für Powerpoint-Dokumente

Für PDF-Dateien brauchen Sie XPDF, einen Parser, der auf einer Programmsuite namens XPDF basiert. Der Parser ist unter den folgenden URLs erhältlich: *http://www.xpdf.com/* (Stand der Quellenangabe: Juni 2009) *und http://www.foolabs.com/xpdf/* . Dazu gehören zwei Programme:

- pdftotext
- pdfinfo

16.5.2 Schnellstartanleitung

Das Einrichten der **imperia** -Suche setzt sich aus den drei Schritten Konfiguration, Indizierung und Template-Erstellung zusammen. Durch die Konfiguration wird unter anderem festgelegt, welche Dokumente über die Suche auffindbar sein sollen. Templates dienen zur Eingabe von Suchbegriffen und Suchparametern sowie zur Darstellung der Suchergebnisse. Bei der Indizierung wird ein Suchindex erstellt. Die eigentliche Suche

beschreibt, in welcher Weise gesuchte Worte miteinander verknüpft werden können. Für alle konfigurierbaren Parameter gibt es Default-Werte. Diese kommen auch in den Konfigurationsbeispielen zum Einsatz. Im Verzeichnis `site/fts/templates` Ihrer **imperia** -Installation liegen Beispiel-Templates für Suchanfragen und die Anzeige der Suchergebnisse. Wenn Sie also auf eine besondere Konfiguration verzichten können, sind nur wenige Schritte notwendig, damit Sie die Volltextsuche einsetzen können:

1. Konfiguration: Im Verzeichnis site/config finden Sie Beispiel-Konfigurationsdateien für die Indexerstellung (`index.conf.sample`) und die Ausgabe der Suche (`fts.conf.sample`). Kopieren Sie diese Dateien nach `index.conf` und `fts.conf`.
2. Indexerstellung: Führen Sie das Script `site/bin/fts_index.pl` mit dem Parametern *–b* und *-i* aus, um den Index zu erstellen.

Hinweis:

Bei umfangreichen Projekten kann dieser Vorgang einige Zeit in Anspruch nehmen.

3. Jetzt können Sie Suchanfragen in die vorbereiteten Templates eingeben, indem Sie das Suchskript `fts_search.pl` aus dem Verzeichnis cgi-bin im Browser aufrufen: *http://ihr_server.de/cgi-bin/fts_search.pl* bzw. *http://ihr_server.de/cgi-bin/fts_search.pl?ADV=1* für ein verfeinertes Suchformular mit zusätzlichen Optionen für die Suche.

16.5.3 Konfiguration der Volltextsuche

Die **imperia** -Volltextsuche gliedert sich in zwei Bereiche - die Erstellung eines Suchindex und die Suche nach Begriffen im Suchindex inklusive der Anzeige der gefundenen Treffer.

Diese funktionale Trennung spiegelt sich auch in der Konfiguration wider, die in zwei getrennten Dateien vorgenommen wird. Bereits bestehende Suchkonfigurationen aus älteren **imperia** -Versionen können mit einem Skript migriert werden. Lesen Sie hierzu Abschnitt 16.5.3.3 **Die Konfiguration der Indexerstellung (index.conf)** auf Seite 182 . Wenn Sie mehrere Indizes verwalten wollen, können Sie dies über die Definition mehrerer Domänen erreichen. Das ermöglicht beispielsweise die Nutzung einer Suchkonfiguration für mehrere Zielsysteme.

Domänen können Sie auch nutzen, um unterschiedliche Ausschnitte eines Datenbestandes getrennt voneinander durchsuchbar zu machen. Wenn Sie beispielsweise einen Teil der auf dem Zielsystem verfügbaren Daten nur für Ihre Mitarbeiter durchsuchbar machen wollen, müssen Sie entsprechende Domänen konfigurieren. In den meisten Fällen reicht es jedoch aus, die Grundeinstellung mit einer Domain ("*default*") zu verwenden.

16.5.3.1 Allgemeine Syntaxkonventionen

Die Syntax der Anweisungen zur Konfiguration der Volltextsuche ist an die Syntax der Konfiguration des Apache Webservers angelehnt, um Ihnen einen schnelleren Einstieg zu ermöglichen. Einige Anweisungen werden in spitzen Klammern notiert. Diese "Tags" bilden Blöcke, in denen Sie weitere Tags oder Direktiven notieren können, um beispielsweise eine Reihe von Parametern für Verzeichnisse festzulegen. Ähnlich wie bei Markup-Sprachen oder auch bei der Apache-Konfiguration gibt es dabei ein einleitendes und ein abschließendes Tag.

Beispiel:

```
<option>
 Direktiven
</option>
```

Bei manchen Tags wird im einleitenden Tag bereits ein Parameter festgelegt. Dieser kann durch ein optionales Gleichheitszeichen und durch Anführungsstriche, deren Verwendung ebenfalls optional ist, besser kenntlich gemacht werden. Beispiel:

```
<directory = "pfad/zum/verzeichnis">
 Direktiven
</directory>
```

bzw.

```
<directory pfad/zum/verzeichnis>
 Direktiven
</directory>
```

Außerdem gibt es noch Direktiven, die in der Form

```
Direktive="Wert"
```

notiert werden und einen Wert für einen bestimmten Parameter bestimmen.

Beispiel:

```
chmap = "UTF-8"
```

Auch hier ist Ihnen die Verwendung von Gleichheitszeichen und Anführungszeichen freigestellt. Eine Ausnahme bilden sowohl für Tags als auch für einfache Direktiven die Fälle, in denen der beschriebene Parameter bzw. der gesetzte Wert Leerzeichen enthält. Dort ist die Verwendung von Anführungszeichen bindend, um sicherzustellen, dass der komplette Ausdruck gewertet wird.

Allgemein ist der Einsatz von Anführungszeichen jedoch empfehlenswert, um Fehlinterpretationen zu vermeiden. Das oben stehende Beispiel würde nämlich mit einer winzigen Änderung nicht mehr funktionieren.

Falsch ist:

```
<directory pfad/zum/verzeichnis >
 Direktiven
</directory>
```

Das Leerzeichen vor der schließenden spitzen Klammer interpretiert der Parser als Bestandteil der Pfadangabe, also "pfad/zum/verzeichnis " statt "pfad/zum/verzeichnis". Im Gegensatz dazu funktioniert die Variante mit Anführungsstrichen problemlos.

Richtig:

```
<directory "pfad/zum/verzeichnis" >
 Direktiven
</directory>
```

In diesem Fall ignoriert der Parser das überflüssige Leerzeichen einfach. Bei den Tags und Direktiven der Konfigurationsdateien wird Groß- und Kleinschreibung grundsätzlich nicht berücksichtigt. Die Codebeispiele in den folgenden Abschnitten verwenden allerdings durchgängig die Kleinschreibweise bzw. die "CamelCase"-Schreibweise.

Kommentarzeilen leiten Sie durch ein Raute-Zeichen (#) am Zeilenanfang ein. Verwenden Sie für Wertzuweisungen keine Umlaute oder Sonderzeichen außer Bindestrich (-) und Unterstrich (_). Ausgenommen hiervon sind reguläre Ausdrücke.

16.5.3.2 Reguläre Ausdrücke in der Konfiguration

Reguläre Ausdrücke in den Konfigurationsdateien sind in ihrer Funktionalität stark an Perl angelehnt. Ihnen stehen sämtliche syntaktischen Möglichkeiten, die Sie in Perl im "Rumpf" eines Suchmusters (zwischen den Schrägstrichen) notieren können, zur Verfügung. Modifier (single line, multi line, ignore case etc.) sind nicht verwendbar. Generell werden die von Ihnen definierten Muster mit der Option "ignore case" ausgewertet. Einen Sonderfall stellen Datei- bzw. Verzeichnisnamen dar. Hier ist je nach Betriebssystem die Groß- und Kleinschreibung relevant.

16.5.3.3 Die Konfiguration der Indexerstellung (index.conf)

Über die Datei `site/config/index.conf` konfigurieren Sie die Indexerstellung. Die Konfiguration wird vom Skript `fts_index.pl` eingelesen und verarbeitet.

Wichtig:

Wichtig: Änderungen an der Konfiguration der Indexerstellung werden erst aktiv, wenn Sie einen neuen Index erstellen.

Im Folgenden finden Sie eine Auflistung der verwendeten Tags und Direktiven. Die einzelnen Anweisungen sind dabei jeweils in der höchstmöglichen Strukturebene aufgeführt, in der sie verwendbar sind. Einige können Sie auch noch an anderen Stellen einsetzen, was dann entsprechend vermerkt ist.

165331 Basis-Tags und -Direktiven

<domain>

Das Tag „domain“ beschreibt verschiedene unabhängige Konfigurationen für verschiedene Indizes innerhalb einer Konfigurationsdatei. Bis auf wenige Ausnahmen, die allgemeingültige Einstellungen festlegen und deswegen auch auf oberster Ebene definiert werden können, notieren Sie alle weiteren Tags und Direktiven innerhalb eines Domain-Abschnittes.

Die Verwendung von Domains ist flexibel. Sie können entweder verschiedene Datenbestände oder auch ein- und denselben Datenbestand mit unterschiedlichen Einstellungen indizieren. Beispielsweise lassen sich unterschiedliche Domains für Mitarbeiter und Kunden einrichten. Bei der Indexerstellung für die Suche, die von Ihren Kunden genutzt werden soll, würden Sie dann die Bereiche mit Informationen ausklammern, die nur zum firmeninternen Gebrauch bestimmt sind. Der Index für die von den Mitarbeitern genutzte Suche würde jedoch auch die nicht-öffentlichen Bereiche einschließen.

Gegenüber einer Filterung, die Sie zum Beispiel durch Gruppierung der Information erreichen, hat dies den Vorteil, dass die Trennung der verfügbaren Daten sicherer ist. In Informationsgruppen „versteckte“ Inhalte sind demgegenüber durch geschickte bzw. glückliche Manipulation der Aufrufparameter des Suchskripts unter Umständen auch unberechtigten Besuchern zugänglich.

Hinweis:

Die in der Datei index.conf.sample genannte domain „default“ ist eine eigenständige Domain und dient nicht etwa der Erfassung von Default-Werten für andere Domains.

Syntax:

```
<domain= "domain_name">
 [Tags und Direktiven zur Konfiguration des Index dieser Domain]
</domain>
```

Beispiel:

```
<domain = "secure”>
 #limited index only intranet
</domain>
<domain = ”web”>
 #general index public web content
</domain>
```

<settype>

Mit dem Tag „settype“ definieren Sie, welche Dateitypen indiziert werden sollen. Der Aufbau entspricht einer mehrwertigen Liste. Folgende Schlüssel stehen zur Verfügung:

- HTML

  Mit diesem Plug-In können Sie HTML-Dateien indizieren.

- TXT

  Dieses Plug-In ermöglicht die Indizierung von TXT-Dateien.

- XPDF

  Mit Hilfe des XPDF-Plug-Ins ist es möglich, PDF-Dateien zu indizieren.

- DOC

  Um DOC-Dateien zu indizieren, verwenden Sie bitte dieses Plug-In.

- PPT

  Das PPT-Plug-In erlaubt die Indizierung von PPT-Dateien.

- XLS

  Unter Verwendung des XLS-Plug-Ins lassen sich XLS-Dateien indizieren.

- NULL

  Mit Hilfe des NULL-Plug-Ins können Sie einen reinen Meta-Index erstellen.

Syntax:

```
<settype>
 Typ = [regulärer Ausdruck]
</settype>
```

Der reguläre Ausdruck beschreibt Dateiendungen.

Beispiel:

```
<setType>
 HTML = "\..?html?$"
 TXT = "\.txt$"
</setType>
```

Hinweis:

Die Dokumenttypen PDF, DOC, PPT und XLS werden über externe Parser indizierbar gemacht. Dieses Tag kann außerdem innerhalb eines Domain-Abschnittes eingesetzt werden.

Dieses Tag ist auch in Domain-Blöcken anwendbar.

<bonus>

Mit dem Bonus-Tag können Sie die Gewichtung der Volltextsuche beeinflussen, indem Sie einem Begriff abhängig von seiner Position in der Struktur eines Dokuments zusätzliches Gewicht verleihen. Dabei vergeben Sie Boni für bestimmte Strukturelemente und Metafelder eines Dokuments. Diese werden dann bei der Ermittlung der Häufigkeit eines Begriffes in einem Dokument hinzugezählt, wenn ein Begriff innerhalb des entsprechenden Strukturelements vorkommt.

Wenn Sie beispielsweise für Überschriften ersten Grades einen Bonus von zwei eingestellt haben, erhöht sich der Zähler für einen Begriff also jedes Mal um drei, wenn dieser innerhalb einer Überschrift ersten Grades erscheint (+1 für den Begriff, +2 durch den Bonus).

Sie können für folgende Elemente Boni vergeben:

- title (<title></title>)
- bold ()
- italics ()
- header1 (<h1></h1>)
- header2 (<h2></h2>)
- header3 (<h3></h3>)

- header4 (<h4></h4>)
- header5 (<h5></h5>)
- header6 (<h6></h6>)
- keywords (**imperia** -Metafeld oder pageMetaComment `keywords`)
- description (**imperia** -Metafeld pageMetaComment description)

Wichtig:

Die obige Auswahl von Strukturelementen steht Ihnen nur bei HTML-Dokumenten in vollem Umfang zur Verfügung. Bei PDF-Dateien kann lediglich für das Title-Element ein Bonus vergeben werden. Für Microsoft Office®-Dokumente lassen sich keine Boni festlegen.

Als Werte kommen Ganzzahlen und Dezimalzahlen in Frage. Notieren Sie Dezimalzahlen in der englischen Schreibweise mit einem Punkt als Dezimalzeichen.

Syntax:

```
<bonus>
 Strukturelement = Ziffer
</bonus>
```

Beispiel:

```
<bonus>
 italics = 1.2
 title = 2
 bold = 1.3
 header1 = 1.2
 header2 = 1.2
 header3 = 1.2
</bonus>
```

Sie haben außerdem die Möglichkeit, Boni domainspezifisch zu definieren.

<exclude>

Sie können dieses Tag nutzen, um den Inhalt einzelner Metafelder oder Dateien, die ein bestimmtes Metafeld enthalten, von der Indizierung auszuschließen.

Syntax:

```
<exclude>
 [Direktiven]
</exclude>
```

Eine Beschreibung der in Exclude-Blöcken verfügbaren Direktiven und Beispiele finden Sie in Abschnitt 16.5.3.3.3 **Direktiven in Exclude-Blöcken** auf Seite 190. Dieses Tag lässt sich auch innerhalb von Domain-, Directory-, DirectoryMatch- und Files-Blöcken einsetzen.

<include>

Sie können dieses Tag nutzen, um Dateien, die ein bestimmtes Metafeld enthalten, indizieren zu lassen.

Syntax:

```
<include>
 [Direktiven]
</include>
```

Eine Beschreibung der in Include-Blöcken verfügbaren Direktiven und Beispiele finden Sie in Abschnitt 16.5.3.3.4 **Direktiven in Include-Blöcken** auf Seite 190.

Dieses Tag lässt sich auch innerhalb von Domain-, Directory-, DirectoryMatch- und Files-Blöcken einsetzen.

<pageMetaComment>

Mit diesem Tag können Sie ein Metafeld definieren, das durchsucht werden und dessen Inhalt in Templates ausgegeben werden kann. Innerhalb des Tags legen Sie eine Start- und eine Endmarkierung fest, mit denen Sie den Metafeldinhalt vom übrigen Text des Dokuments abgrenzen.

Syntax:

```
<pageMetaComment = "meta_name">
 start = "<!--startcomment-->"
 stop = "<!--stopcomment-->"
</pageMetaComment>
```

Beispiel:

```
<pageMetaComment = "FSK 0">
 start = "<!--start_FSK 0-->"
 stop = "<!--stop_FSK 0-->"
</pageMetaComment>
```

Dieses Tag ist auch in Directory-, DirectoryMatch- und Files-Blöcken verwendbar.

blacklist

Mit dieser Direktive geben Sie eine oder mehrere Stopwortlisten an, die für alle konfigurierten Domains der Volltextsuche gültig sein sollen. In einer Stopwortliste aufgeführte Wörter indiziert die Volltextsuche nicht. Damit eine Suchanfrage die Stopwörter ebenfalls ignoriert, müssen Sie die Direktive auch in der Suchkonfiguration `site/config/fts.conf` setzen.

Syntax:

```
blacklist = BEZEICHNER
```

Mit *BEZEICHNER* geben Sie eine Stopwortliste an, die Sie mit dem Skript `site/bin/fts_blacklist.pl` in das System eingetragen haben. Lesen Sie hierzu auch Abschnitt 16.5.3.7 **Stopwortlisten anlegen und verwalten** auf Seite 213.

Wenn Sie mehrere Stopwortlisten einsetzen möchten, verwenden Sie die Direktive mehrfach, für jede Stopwortliste einmal.

Diese Direktive können Sie auch innerhalb von Domain-Blöcken verwenden, dann gilt die Stopwortliste nur für die betreffende Domain.

PDFINFO

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

PDFTOTEXT

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

XLS2CSV

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

CATDOC

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

CATPPT

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

165332 Tags in Domain-Blöcken

<metaGroup>

Dieses Tag erlaubt Ihnen, Dateien anhand von Metafeldern und deren Inhalt zu Gruppen zusammenzufassen. Diese Gruppen können Sie bei Suchanfragen als Filter an das Suchskript übergeben (lesen Sie hierzu den Abschnitt „Übergabeparameter und Variablen in Suchtemplates“ im Kapitel „Templates für die Volltextsuche“ im „Programmierhandbuch“). In einem MetaGroup-Block definieren Sie eines oder mehrere Metafelder mit zugehörigen Inhalten. Dokumente, die die entsprechenden Metafelder und Metafeldinhalte aufweisen, gehören dann zu dieser Gruppe.

Die eigentliche Definition der einzelnen Metafelder und jeweiligen Gruppen nehmen Sie im Tag *<meta>* vor, das in Abschnitt 16.5.3.3.7 **Tags und Direktiven in MetaGroup-Blöcken** auf Seite 198 beschrieben wird. Dort finden Sie auch eine genauere Erklärung des unten stehenden Beispiels.

Syntax:

```
<metaGroup>
 <meta Metafeldname>
  [Gruppendefinition(en)]
 </meta>
</metaGroup>
```

Beispiel:

```
<metaGroup>
 <meta charset>
  <groupname "CZECH">
   match = "ISO-8859-2"
  </groupname>
  <groupname "RU">
   match = "ISO-8859-5|KOI.+"
  </groupname>
  <groupname "INTL">
   match = "UTF-8"
  </groupname>
  <groupname "WESTERN">
   match = "ISO-8859-1"
  </groupname>
 </meta>
</metaGroup>
```

Meta-Gruppen lassen sich außerdem noch innerhalb von Directory-Blöcken einrichten.

<option>

Das Option-Tag beschreibt verschiedene Direktiven zur Steuerung der Indexerstellung. Die Beschreibung der einzelnen Direktiven finden Sie in Abschnitt 16.5.3.3.6 **Option-Direktiven** auf Seite 195.

Syntax:

```
<option>
 [Direktiven]
</option>
```

Beispiel:

```
<option>
 verbosityLevel = 1
 followSymLinks = ON
</option>
```

<directory>

Das Directory-Tag dient in erster Linie zur Festlegung der Verzeichnisse, die indiziert werden sollen. Außerdem können Sie es als Container für weitere Parameter nutzen. Die Indizierung des documentDir, die der `DOCUMENT-ROOT` Ihres Systems entspricht, ist per Default aktiviert.

Syntax:

```
<directory "pfad/zum/verzeichnis”>
 [Direktiven]
</directory>
```

Absolute Pfadangaben notieren Sie mit einem führenden Slash bzw. einem Laufwerksbuchstaben gefolgt von einem Doppelpunkt. Alle anderen Pfadangaben werden relativ zum `documentDir` ausgewertet.

Tipp

Wenn Sie die Indizierung des documentDir deaktivieren wollen, notieren Sie folgende Directory-Definition:

```
<directory "">
 index = 0
</directory>
```

Beispiele:

Beim folgenden Beispiel handelt es sich um eine absolute Pfadangabe. Das Verzeichnis entspricht dabei dem Default-Wert für die Direktive `documentDir` (siehe Abschnitt 16.5.3.3.6 **Option-Direktiven** auf Seite 195). Alternativ könnten Sie auch mit einem Leerstring (<directory="">) darauf verweisen.

```
<directory /imperia/htdocs>
 [Direktiven]
</directory>
```

Das folgende Beispiel zeigt eine relative Pfadangabe. Das Verzeichnis `FTS` befindet sich direkt unterhalb des documentDir-Verzeichnisses.

```
<directory FTS>
 [Direktiven]
</directory>
```

Die Beschreibung der einzelnen Direktiven finden Sie in Abschnitt 16.5.3.3.5 **Tags und Direktiven in Directory-Blöcken** auf Seite 191.

<directoryMatch>

Wenn Sie statt eines Verzeichnisnamens Bestandteile des Verzeichnisnamens als Kriterium für die Anwendung bestimmter Direktiven einsetzen wollen, nutzen Sie das Tag *<directoryMatch>*. Im einleitenden Tag definieren Sie einen regulären Ausdruck. Dieser wird bei der Indexerstellung auf die einzelnen Verzeichnisse angewendet. Passt der reguläre Ausdruck auf den Verzeichnisnamen, kommen die im entsprechenden DirectoryMatch-Block definierten Direktiven zur Anwendung.

Hinweis:
Es werden nacheinander nur die einzelnen Verzeichnisnamen geprüft. Es ist nicht möglich, auf Bestandteile des gesamten Verzeichnispfades über Verzeichnistrenner hinweg zu prüfen.

Syntax:

```
<directoryMatch = "[regulärer_Ausdruck]">
 [Direktiven]
</directoryMatch>
```

Beispiel:

```
<directoryMatch = ".+print$">
 index = 0
</directoryMatch>
```

Nach dieser Definition würden folgende Verzeichnisse nicht indiziert:

- `nachrichten_print`
- `produkte_print`

Trotzdem indiziert würden jedoch folgende Verzeichnisse:

- `nachrichten_print/doc`
- `nachrichten_print/pdf`
- `produkte_print/doc`
- `produkte_print/pdf`
- `print`

Dieses Beispiel soll verdeutlichen, dass die Direktiven immer nur auf das Verzeichnis angewendet werden, dessen Name auf den regulären Ausdruck passt. Liegt ein passender Verzeichnisname im Pfad eines Verzeichnisses, haben die Direktiven dafür keine Gültigkeit.

<files>

Mit dem Files-Tag können Sie Direktiven für bestimmte Dateien festlegen. Im einleitenden Tag definieren Sie dabei mit einem regulären Ausdruck, welche Dateien beachtet werden. Die Direktiven werden auf Dateien angewendet, deren Name auf das Muster passt.

Syntax:

```
<files = "regulärer Ausdruck">
 [Direktiven]
</files>
```

Beispiele:

```
<files = "parse_this.html">
 index = 2
</files>
```

Die Datei `parse_this.html` wird indiziert.

Hinweis:

Die Direktive index=2 aktiviert die Indizierung. Der Wert 2 wird innerhalb von files-Blöcken wie 1 bzw. yes bzw. true gewertet. Lesen Sie hierzu auch Abschnitt 16.5.3.3.5 **Tags und Direktiven in Directory-Blöcken** *auf Seite 191.*

```
<files \.text>
 index = 1
 chmap = "UTF-8"
</files>
```

Dateien mit der Endung `.text` werden indiziert. Für Erläuterungen zu den in den Beispielen verwendeten Direktiven lesen Sie bitte Abschnitt 16.5.3.3.5 **Tags und Direktiven in Directory-Blöcken** auf Seite 191.

Dieses Tag können Sie auch in Directory- und DirectoryMatch-Blöcken einsetzen.

165333 Direktiven in Exclude-Blöcken

fileMetaKey

Bestimmen Sie ein Metafeld, bei dessen Vorhandensein das betreffende Dokument nicht indiziert werden soll. Der Wert dieser Direktive wird als regulärer Ausdruck gewertet.

Syntax:

```
fileMetaKey = "regulärer_Ausdruck"
```

metaKey

Geben Sie ein Metafeld an, dessen Inhalt nicht indiziert werden soll. Der Wert dieser Direktive wird als regulärer Ausdruck gewertet.

Syntax:

```
metaKey = "regulärer_Ausdruck"
```

Beispiel für einen Exclude-Block mit Direktiven:

```
<exclude>
 fileMetaKey "^mod.*"
 fileMetaKey "^layout_test$"
 metaKey "imperia"
</exclude>
```

Das Beispiel legt folgende Ausschlusskriterien fest:

- Dateien, die ein Metafeld enthalten, dessen Name mit "mod" beginnt, wobei dahinter weitere Zeichen folgen können, werden nicht indiziert.
- Dateien mit dem Metafeld *layout_test* werden nicht indiziert.
- Inhalte von Metafeldern, die im Namen die Zeichenfolge "imperia" enthalten, werden nicht in den Index aufgenommen.

165334 Direktiven in Include-Blöcken

fileMetaKey

Bestimmen Sie ein Metafeld, bei dessen Vorhandensein das betreffende Dokument indiziert werden soll. Der Wert dieser Direktive wird als regulärer Ausdruck gewertet.

Syntax:

```
fileMetaKey = "regulärer_Ausdruck"
```

Beispiel für einen Include-Block mit Direktiven:

```
<exclude>
 fileMetaKey "^mod.*"
 fileMetaKey "^layout_test$"
</exclude>
```

Das Beispiel definiert folgende Metafelder als Kriterium für ein Indizieren eines Dokuments:

- Dateien, die ein Metafeld enthalten, dessen Name mit "mod" beginnt, wobei dahinter weitere Zeichen folgen können, werden indiziert.
- Dateien mit dem Metafeld *layout_test* werden indiziert.

165335 Tags und Direktiven in Directory-Blöcken

<pageMeta>

Mit diesem Tag bestimmen Sie, welche Metataginhalte und **imperia** -Metafeldinhalte in den Index aufgenommen werden und unter welchem Namen sie dort referenzierbar sind.

Syntax:

```
<pageMeta>
 Name_des_imperia_Metafelds = meta_alias_im_Suchtemplate
 Name_des HTML-Metatags = meta_alias_im_Suchtemplate
</pageMeta>
```

Beispiel:

In PDF-Dateien sind neben dem eigentlichen Inhalt auch einige Metainformationen gespeichert. Allerdings liegen diese als Textinformationen in einem Format vor, auf das Sie nicht ohne Weiteres zugreifen können. Die Direktive pageMeta ermöglicht Ihnen, gezielt die Metainformationen eines PDF-Dokuments zu indizieren, um Sie bei der Ausgabe eines Suchergebnisses verwenden zu können. Das Programm `pdfinfo` liefert folgende Metainformationen zu einem PDF-Dokument:

```
Title:     Microsoft Word - MyTitle
Author:     John Doe
Creator:    PScript5.dll Version 5.2.2
Producer:    Acrobat Distiller 6.0 (Windows)
CreationDate:  Fri Jul 8 10:28:06 2005
ModDate:    Tue Aug 9 11:18:17 2005
Tagged:     no
Pages:     23
Encrypted:   no
Page size:   595 x 842 pts (A4)
File size:   343800 bytes
Optimized:   no
PDF version:  1.5
```

Um beispielsweise Titel, Erstellungs- und Änderungsdatum einer PDF-Datei in den Index der Volltextsuche aufzunehmen, definieren Sie folgenden PageMeta-Block:

```
<pagemeta>
 title = dc_title
 creationdate = page_time
 moddate = mod_date
</pagemeta>
```

Die indizierten Meta-Tag- bzw. **imperia** -Metafeld-Inhalte lassen sich im Ergebnis-Template ausgeben. Enthält ein Dokument in der Trefferliste die indizierten Meta-Tags oder Metafelder, erscheint der jeweilige Inhalt in an der entsprechenden Stelle in der Trefferliste.

Enne Suchanfrage nach den Metafeld- bzw. Meta-Tag-Inhalten liefert ebenfalls Treffer. Auch, wenn die Inhalte nicht zum sichtbaren Text des Dokuments gehören, wie das zum Beispiel bei Meta-Tags der Fall ist.

Einzelheiten zur Ausgabe von in PageMeta-Blöcken indizierten Daten finden Sie im Kapitel "**Templates für die Volltextsuche**" im Programmierhandbuch.

Das PageMeta-Tag können Sie auch innerhalb von DirectoryMatch- und Files-Blöcken einsetzen.

dirGroup

Die Direktive `dirGroup` ermöglicht es, Verzeichnisse bzw. Verzeichnisinhalte zu Gruppen zusammenzufassen. Ein Verzeichnis kann dabei mehreren Gruppen angehören. So definierte Gruppen lassen sich mit dem Parameter *GROUP* beim Aufruf des Suchskripts zur Filterung verwenden (Lesen Sie hierzu auch den Abschnitt Übergabeparameter und Variablen in Suchtemplates im Kapitel "**Templates für die Volltextsuche**" im Programmierhandbuch).

Syntax:

```
dirGroup = [Gruppenname(n)]
```

Beispiel:

```
<directory ="produkte_de">
 index = yes
 dirGroup = german
</directory>
<directory ="news_de">
 index = yes
 dirGroup = german
</directory>
<directory = "produkte_en">
 index= yes
 dirGroup = english
</directory>
<directory = "produkte_en">
 index= yes
 dirGroup = english
</directory>
```

Die Verzeichnisse `produkte_de` und `news_de` gehören nach oben stehender Definition zur Gruppe `german`, die Verzeichnisse `produkte_en` und `news_en` zur Gruppe `english`. Das gilt ebenfalls für die jeweiligen Unterverzeichnisse. Beispielsweise gehören auch die Verzeichnisse `produkte_de/camping` und `produkte_de/haushalt` zur Gruppe `german`.

Mehrere Gruppen notieren Sie durch Leerzeichen voneinander getrennt.

Beispiel:

```
<directory ="produkte_de">
 index = yes
 dirGroup = sortiment german produkte
</directory>
```

Diese Direktive ist auch innerhalb von DirectoryMatch-Blöcken anwendbar. Beachten Sie dabei die besondere Funktionsweise von *directoryMatch* (siehe Abschnitt 16.5.3.3.2 **Tags in Domain-Blöcken** auf Seite 187).

Beispiel:

```
<directoryMatch =".+_de">
 index = yes
 dirGroup = german
</directory>
<directoryMatch = ".+_en">
```

```
 index= yes
 dirGroup = english
</directory>
```

Alle Verzeichnisse, deren Name auf "_de" endet, gehören zur Gruppe `german`. Alle Verzeichnisse, deren Name auf "_en" endet, gehören zur Gruppe `english`.

Im Unterschied zum vorhergehenden Beispiel schließt das etwaige Unterverzeichnisse nicht mit ein. Die Verzeichnisse `produkte_de/camping` und `produkte_de/haushalt` müssten also `produkte_de/camping_de` und `produkte_de/haushalt_de` benannt sein, um ebenfalls zur Gruppe german gezählt zu werden.

index

Mit der Index-Direktive steuern Sie die Indexerstellung. Sie können einstellen, ob Verzeichnisse bzw. Dateien indiziert werden. Bei Verzeichnissen steuern Sie, ob Unterverzeichnisse ebenfalls indiziert werden sollen. Einstellungen aus übergeordneten Verzeichnissen lassen sich durch Verwendung der Direktive für Unterverzeichnisse (`directory`) oder Dateien (`files`) überschreiben.

Zur Aktivierung einer rekursiv ausgeführten Indizierung setzen Sie die folgenden Werte ein:

- yes
- true
- 1

Um ein Verzeichnis oder eine Gruppe von Dateien von der Indizierung auszuschließen, verwenden Sie folgende Werte:

- no
- false
- 0

Hinweis:

Innerhalb von Directory-Tags gilt „yes“ als Default-Wert, wenn Sie keine Index-Direktive setzen.

Verwenden Sie einen beliebigen anderen Wert, um nur das jeweilige Verzeichnis indizieren zu lassen. Unterverzeichnisse werden dann nicht durchsucht, es sei denn, Sie haben dies durch anderslautende Direktiven explizit aktiviert.

Beispiel:

```
index = yes-no-descend
```

Hinweis:

In Files-Blöcken gibt es diese Option nicht. Dort gilt jeder Wert, der die Option nicht explizit abschaltet, als Aktivierung.

Diese Direktive ist auch innerhalb von DirectoryMatch- und Files-Blöcken anwendbar.

Hinweis:

Innerhalb von Files-Blöcken bezieht sich die Index-Direktive ausschließlich auf die betreffenden Dateien, nicht auf die Verzeichnisse, in denen sich diese befinden.

chmap

Mit dieser Direktive können Sie einen Standard-Zeichensatz für Verzeichnisse festlegen. Bei der Indexerstellung wertet der Parser diese Angabe aus. Gibt es in den indizierten Dateien eine abweichende Angabe, z. B. durch ein entsprechendes Meta-Tag, hat diese Vorrang. Als mögliche Werte kommen alle vom Perlmodul *Locale::Recode* unterstützten Zeichensätze in Frage.

Syntax:

```
chmap = "Zeichensatz"
```

Beispiel:

```
chmap= "ISO-8859-1"
```

Diese Direktive können Sie auch in DirectoryMatch-Blöcken verwenden.

inherit

Diese Direktive ermöglicht es Ihnen, Einstellungen für ein Verzeichnis an dessen Unterverzeichnisse weiterzuvererben, auch wenn Sie das Verzeichnis selbst nicht bzw. nicht rekursiv indizieren lassen wollen. Einstellungen sind mit *inherit* nur über eine Verzeichnisebene vererbbar. In Unterverzeichnissen gesetzte, widersprüchliche Direktiven überschreiben vererbte Einstellungen.

Hinweis:

Die Einstellung für „index“ lässt sich nicht vererben.

Syntax:

```
inherit
```

Beispiel:

```
<directory = "news_de">
 index = "0"
 chmap = "ISO-8859-1"
 inherit
</directory>

<directory = "news_de/wirtschaft">
 index = "1"
</directory>

<directory = "news_de/politik">
 index = "1"
</directory>

<directory = "news_de/sport">
 index = "not-recursive"
 inherit
</directory>

<directory = "news_de/wetter">
 index = "1"
 chmap = "utf-8"
</directory>

<directory = "news_de/sport/fliegenfischen">
 index = "1"
</directory>
```

```
<directory = "news_de/sport/eisstockschiessen">
 index = "1"
</directory>
```

In diesem Beispiel soll das Verzeichnis `news_de` nicht indiziert werden. Allerdings wird festgelegt, dass der Inhalt ISO-8859-1-encodiert ist. Diese Einstellung soll auf Unterverzeichnisse weitervererbt werden, weswegen die Direktive inherit gesetzt wird. In den Unterverzeichnissen `news_de/wirtschaft`, `news_de/politik` und `news_de/sport` gilt deswegen auch die in der übergeordneten Ebene gesetzte chmap-Direktive.

Ausnahme ist das Verzeichnis `news_de/wetter`, da hier mit *chmap = "utf-8"* ein anderer Zeichensatz eingestellt wird. Da das Verzeichnis `news_de/sport` wegen der Einstellung *index = "not-recursive"* auch nicht rekursiv durchsucht wird, muss mit einem weiteren Einsatz von *inherit* dafür gesorgt werden, dass die beiden Verzeichnisse `news_de/sport/fliegenfischen` und `news_de/sport/eisstockschiessen` ebenfalls als ISO-8859-1-encodiert gelten. Ohne diese weitere Inherit-Direktive würde die Zeichensatzeinstellung nicht weitervererbt.

Inherit lässt sich auch in Exclude-, Files-, MetaGroup-, PageMetaComment- und PageMeta-Blöcken einsetzen, wenn diese innerhalb eines Directory-Blocks stehen.

165336 Option-Direktiven

cacheSize

Diese Direktive gibt die Anzahl der Begriffe an, die in einer Cache-Datei enthalten sind. Wenn Sie sie auf den Wert 0 setzen oder entfernen, wird kein Cache erstellt. Werte kleiner als 100 sind nicht sehr empfehlenswert, da dann sehr kleine Dateien generiert werden und so je nach Größe der Zuordnungseinheiten Speicherplatz auf der Festplatte verschwendet wird.

Syntax:

```
cacheSize = [Zahl]
```

Beispiel:

```
cacheSize = 150
```

compress

Mit dieser Direktive wählen Sie das Kompressions-Plug-In aus. Um eine Liste der verfügbaren Plug-Ins zu erhalten, rufen Sie das Skript `site/bin/fts_index.pl` mit dem Parameter *-t* in Ihrer **imperia** -Installation auf.

Hinweis:

Ein Wechsel des Kompressions-Plug-Ins erfordert möglicherweise auch eine Anpassung der Einstellung cacheSize.

Der Default-Wert ist 0 (kein Kompressions-Plug-In).

cacheDir

Mit dieser Direktive geben Sie das Verzeichnis an, in dem der Datei-Cache der Volltextsuche gespeichert wird.

Syntax:

```
cacheDir = "/absoluter/pfad"
```

Beispiel:

```
cacheDir = "/usr/local/share/imperia/site/fts/index/mandant1/cache"
```

Wichtig:

Wenn Sie diese Variable setzen, müssen Sie auch darauf achten, dass das Verzeichnis nur von einem Index bzw. für eine Domain genutzt wird. Unter Umständen, zum Beispiel wenn mehrere Domains auf sich überschneidende Datenbestände zugreifen, kann es jedoch sogar wünschenswert sein, dass zumindest Teile des Caches von den Domains gemeinsam genutzt werden. Bitte wenden Sie sich im Zweifelsfall an den **imperia** *-Support.*

Das System legt für jede konfigurierte Domain automatisch ein Unterverzeichnis an, in dem dann der eigentliche Cache gespeichert wird.

Wichtig:

Dies gilt nur, wenn Sie den Default-Wert nutzen.

Der Default-Wert ist `dataDir/cache`.

dataDir "/absoluter/pfad"

Tragen Sie hier den absoluten Pfad ein, unter dem der erzeugte Index abgelegt wird. Bei der Indexerstellung wird automatisch für jede konfigurierte Domain ein eigenes, nach der jeweiligen Domain benanntes Verzeichnis unterhalb dieser *dataDir* angelegt. Der eigentliche Index liegt in diesem Unterverzeichnis.

Syntax:

```
dataDir = "/absoluter/pfad"
```

Beispiel:

```
dataDir = "/usr/local/share/imperia/site/fts/index/mandant1"
```

Der Default-Wert ist das Verzeichnis `site/fts/index` Ihrer **imperia** -Installation. Dieser Wert muss in der Konfiguration der Indexerstellung (Datei `index.conf`) und der Suche (Datei `fts.conf`) gleich sein.

dataTempDir

Mit dieser Direktive legen Sie das temporäre Arbeitsverzeichnis für die Indexerstellung fest. Tragen Sie eine absolute Pfadangabe als Wert ein. Die temporären Dateien der Indexerstellung werden für jede konfigurierte Domain in einem nach der Domain benannten Unterverzeichnis abgelegt.

Syntax:

```
dataTempDir = "/absoluter/pfad"
```

Beispiel:

```
dataTempDir = "/usr/local/share/imperia/site/fts/index/tmp"
```

Der Default-Wert ist `site/fts/index/tmp`.

documentDir

Definieren Sie einen absoluten Pfad zum Verzeichnis, das die zu indizierenden Dokumente enthält.

Syntax:

```
documentDir = "/absoluter/pfad"
```

Beispiel:

```
documentDir = "/usr/local/share/imperia/htdocs"
```

Der Default-Wert ist das **imperia** -Document-Root.

followSubDir

Mit dieser Direktive können Sie die rekursive Indizierung aktivieren bzw. deaktivieren.

```
followSubDir = [ON|OFF]
```

Die Default-Einstellung ist *ON*.

followSymlink

Mit dieser Direktive regeln Sie die Behandlung von symbolischen Links.

Syntax:

```
followSymlink = [ON|OFF]
```

Per Default ist diese Option aktiviert.

maxDepth

Mit dieser Direktive steuern Sie die maximale Verzeichnistiefe, die bei der Indexerstellung berücksichtigt wird.

Syntax:

```
maxDepth = [Zahl]
```

Der Default-Wert ist 100 (Verzeichnisebenen).

memUseControl numerischer Wert

Setzen Sie hier einen numerischen Wert ein, der die maximale Größe der Speicherbereiche für die Indexerstellung festlegt. Mit dieser Direktive beeinflussen Sie die Performance und allgemeine Nutzung des Arbeitsspeichers.

Syntax:

```
memUseControl = [Zahl]
```

Der Default-Wert ist 6400000 (Bytes).

verbosityLevel

Mit dieser Direktive kontrollieren Sie die Menge der Informationen, die während der Indexerstellung ausgegeben werden.

Syntax:

```
verbosityLevel = [0-3]
```

Mögliche Werte:

Wert	Beschreibung
0	"Minimal", nur die jeweiligen Schritte der Indexerstellung werden angezeigt.
1	"Normal", jeweilige Schritte der Indexerstellung und indizierte Dateien sowie etwaige Fehler bei der Erstellung der Dateiliste oder beim Parsing werden ausgegeben.
2	"Ausführlich", zusätzlich zu den normalen Meldungen werden ausgelassene Verzeichnisse und Statistiken der SR-Dateien ausgegeben.
3	Wie "ausführlich", zusätzlich werden ausgelassene Dateien in den verarbeiteten Verzeichnissen und eine Liste der momentan indizierten Verzeichnisse ausgegeben.

Tabelle 16.2. Einstellungen der Direktive *verbosityLevel*

165337 Tags und Direktiven in MetaGroup-Blöcken

<meta>

Dieses Tag nutzen Sie, um innerhalb von MetaGroup-Blöcken Gruppen für einzelne Metafelder festzulegen. Folgende Metafelder sind für jedes Dokument verfügbar:

Feld	Beschreibung
charset	Dies beschreibt den Zeichensatz, in dem das Dokument ursprünglich encodiert ist - nicht zu verwechseln mit dem in der **imperia** -internen Datenhaltung verwendeten Zeichensatz (UTF-8).
filename	Dies ist der Dateiname des Dokuments.
title	Dies ist der Dokumententitel, es sei denn, dieses Feld wurde durch entsprechende Konfiguration explizit geleert.

Tabelle 16.3. allgemein verfügbare Metafelder

Hinweis:

Bei Dokumenten, die nicht mit **imperia** *erzeugt wurden, zum Beispiel außerhalb* **imperia** *s generierten PDF-Dateien oder Microsoft-Office Dokumenten, ist das Feld* `title` *leer.*

Syntax:

```
<meta = "Metafeldname">
 [Gruppendefinition(en)]
</meta>
```

Gruppen lassen sich für jedes in einem Dokument verfügbare Metafeld definieren. Den Namen des betreffenden Feldes notieren Sie im einleitenden Tag eines Meta-Blocks. Anschließend können Sie für das Metafeld eine oder mehrere Gruppen definieren. Dabei legen Sie den Namen einer Gruppe mit dem Tag *groupname* fest.

<groupname>

Den Namen einer Meta-Gruppe legen Sie mit diesem Tag fest. Gruppennamen sind frei wählbar, dürfen aber keine Leerzeichen oder Sonderzeichen außer Binde- und Unterstrich enthalten. Geben Sie statt eines feststehenden Namens einen regulären Ausdruck an, um dynamisch benannte Gruppen zu erhalten. Lesen Sie hierzu auch Abschnitt 16.5.3.3.8 **Dynamisch benannte Meta-Gruppen** auf Seite 199. Innerhalb des Tags definieren Sie ein Suchmuster für Metafeldinhalte.

Syntax:

```
<groupname "String | regulärer Ausdruck">
 [Suchmuster]
</groupname>
```

Das Suchmuster definieren Sie mit der folgenden Direktive.

match

Legen Sie einen regulären Ausdruck fest, der während der Indexerstellung auf das betreffende Metafeld eines Dokuments angewendet wird. Passt das in dieser Direktive festgelegte Muster auf den Inhalt des betreffenden Metafeldes, gehört das Dokument zu der entsprechenden Gruppe.

Beispiel für eine Definition von Metagruppen:

```
<metaGroup>
 <meta charset>
  <groupname "CZECH">
   match = "ISO-8859-2"
  </groupname>
  <groupname "RU">
   match = "ISO-8859-5|KOI.+"
  </groupname>
  <groupname "INTL">
   match = "UTF-8"
  </groupname>
  <groupname "WESTERN">
   match = "ISO-8859-1"
  </groupname>
 </meta>
</metaGroup>
```

In diesem Beispiel werden für das Metafeld *charset* folgende vier Metagruppen definiert:

- *CZECH*, alle Dokumente, deren Charset "ISO-8859-2", bzw. "iso-8859-2" enthält
- *RU*, alle Dokumente, deren Charset "ISO-8859-5", bzw. "iso-8859-5" oder "KOI" bzw. "koi" gefolgt von beliebigen Zeichen enthält
- *INTL*, alle Dokumente, deren Charset "UTF-8", bzw. "utf-8" enthält
- *WESTERN*, alle Dokumente, deren Charset "ISO-8859-1", bzw. "iso-8859-1" enthält

Hinweis:

Bei der Auswertung regulärer Ausdrücke wird die Groß- und Kleinschreibung grundsätzlich nicht berücksichtigt. Lesen Sie hierzu auch Abschnitt 16.5.3.2 **Reguläre Ausdrücke in der Konfiguration** *auf Seite 182.*

165338 Dynamisch benannte Meta-Gruppen

Es ist auch möglich, Gruppennamen um einen dynamischen Bestandteil zu ergänzen. Dieser dynamische Bestandteil ergibt sich aus dem Metafeldinhalt, der die Gruppenzugehörigkeit bestimmt. So können Sie dynamische Gruppennamen zusammensetzen. Auf diese Weise definierte Gruppen sind eigenständig. Die Syntax zur Erzeugung dynamischer Gruppennamen basiert auf der Funktionsweise der regulären Ausdrücke in Perl:

```
<metaGroup>
 <meta Metafeldname>
  <groupname "Gruppenname_${1}">
   match = "(muster1)"
   match = "(muster2)"
  </groupname>
 </meta>
</metaGroup>
```

Damit die regulären Ausdrücke als Suffix an den Namen der Gruppe angehängt werden können, müssen Sie die Suchmuster in Klammern setzen (Gruppierung in regulären Ausdrücken). Dann kann der Zugriff auf die Gruppe im Suchmuster mit *${1}* erfolgen.

Beispiel:

```
<metaGroup>
 <meta title>
  <groupname "planet-${1}">
   match = "(weather)"
   match = "(forecast)"
   match = "THIS IS NOT WHAT YOU WANT"
  </groupname>
 </meta>
</metaGroup>
```

Diese Definition legt anhand des Inhalts des Metafelds *title* drei Gruppen fest:

- Dokumente mit dem Wort "weather" im title gehören zur Gruppe "planet-weather".
- Dokumente mit dem Wort "forecast" im title gehören zur Gruppe "planet-forecast".
- Dokumente mit dem Satz "THIS IS NOT WHAT YOU WANT" im title gehören zur Gruppe "planet-".

Die dritte Gruppe ist zu Demonstrationszwecken falsch notiert und liefert nicht das (wahrscheinlich) erwünschte Ergebnis. Bei der Match-Direktive fehlen die Klammern, daher ergibt sich aus dem Suchmuster und dem feststehenden Teil des Namens kein dynamischer Gruppenname. Außerdem enthält das Muster Leerzeichen, und diese sind in Gruppennamen nicht erlaubt.

16.5.3.4 Die Konfiguration der Suche (fts.conf)

Mit den Einstellungen in der Datei `fts.conf` konfigurieren Sie, wie die Volltextsuche im Index nach eingegebenen Suchbegriffen sucht. Sollten Sie bereits eine ältere Version der **imperia** -Volltextsuche eingesetzt haben, bietet Ihnen das Skript `fts_conf_convert.pl` aus dem Verzeichnis `site/bin` Ihrer **imperia** -Installation die Möglichkeit, Ihre alte Suchkonfiguration in das Format der neuen Volltextsuche zu konvertieren. Die notwendigen Dateien werden dabei automatisch angelegt.

Eine Beispieldatei mit der notwendigen Minimalkonfiguration finden Sie im Verzeichnis `site/config` unter dem Namen `fts.conf.sample`. Sämtliche Parameter verfügen über Default-Werte, die auch in der Beispieldatei zum Einsatz kommen.

Hinweis:

Durch die Default-Werte können Sie die Suche auch ohne die Datei `fts.conf` *bzw. mit einer leeren Datei nutzen. Allerdings sollten Sie dann vorher prüfen, ob die Default-Konfiguration für Ihre Zwecke genügt.*

16.5.3.4.1 Basis-Tags

blacklist

Mit dieser Direktive geben Sie eine oder mehrere Stopwortlisten an, die für alle konfigurierten Domains der Volltextsuche gültig sein sollen. Die Suche ignoriert in einer Stopwortliste aufgeführte Wörter in Suchbegriffen. Damit die Volltextsuche die Stopwörter auch nicht indiziert, müssen Sie die Direktive auch in der Indexkonfiguration `site/config/index.conf` setzen.

Syntax:

```
blacklist = BEZEICHNER
```

Mit *BEZEICHNER* geben Sie eine Stopwortliste an, die Sie mit dem Skript `site/bin/fts_blacklist.pl` in das System eingetragen haben. Lesen Sie hierzu auch Abschnitt 16.5.3.7 **Stopwortlisten anlegen und verwalten** auf Seite 213.

Wenn Sie mehrere Stopwortlisten einsetzen möchten, verwenden Sie die Direktive mehrfach, für jede Stopwortliste einmal.

Diese Direktive können Sie auch innerhalb von Domain-Blöcken verwenden, dann gilt die Stopwortliste nur für die betreffende Domain.

<domain>

Die Einstellungen für die Suche innerhalb eines Index fassen Sie mit dem Tag `domain` in einem Block zusammen. In der Datei `fts.conf` lassen sich mehrere dieser Blöcke einsetzen. Sie können also mehrere Sets von unterschiedlichen Sucheinstellungen in einer Datei verwalten.

Diese Sets können sich sowohl auf einen einzigen als auch auf mehrere verschiedene Datenbestände beziehen. Jedes Set bzw. jede konfigurierte Domain benötigt ein eigenes Suchskript (siehe auch Abschnitt 16.5.3.6 **Erstellung von Suchskripten** auf Seite 213).

Alle weiteren Einstellungen werden innerhalb eines Domain-Blocks notiert.

Syntax:

```
<domain = "domain_name">
 [Direktiven]
</domain>
```

Der Default-Wert für den Domainnamen ist "*default*" bzw. ein Leerstring.

165342 Tags in Domain-Blöcken

<option>

Das Option-Tag fasst alle indexbezogenen Direktiven zu einem Block zusammen.

Syntax:

```
<option>
 [Direktiven]
</option>
```

Die Erklärung zu den Direktiven in Option-Blöcken und Beispiele finden Sie unter Abschnitt 16.5.3.4.3 **Direktiven in Option-Blöcken** auf Seite 204.

<lang>

Innerhalb eines Lang-Blocks definieren Sie sprachspezifische Synonyme für Bestandteile eingegebener Suchbegriffe. So lassen sich etwa die logischen Operatoren zur Verknüpfung mehrerer Suchwörter in unterschiedliche Sprachen übersetzen. Mit dem CGI-Parameter *lang* übergeben Sie beim Absenden von Suchanfragen die gewünschte Sprachversion, was dann bei der Verarbeitung der Anfrage die in der Suchkonfiguration hinterlegten Übersetzungen aktiviert.

Syntax:

```
<lang "Sprachkürzel">
  Wort = Übersetzung
</lang>
```

Beispiel:

```
<lang "DE">
  or = oder
  not = nicht
  and = und
</lang>
```

Dieses Beispiel zeigt die Ersetzung der logischen Operatoren *OR*, *NOT* und *AND* durch ihre deutschen Übersetzungen, wie sie auch in der Standard-Suchkonfiguration enthalten ist. Schicken Sie nun eine Suchanfrage mit dem CGI-Parameter *lang=de*, bzw. *lang=DE* ab, interpretiert die Volltextsuche die Wörter **oder**, **nicht**, **und** als logische Parameter, wenn sie im Suchbegriff vorkommen.

Hinweis:

Dies gilt auch, wenn Sie **oder**, **nicht**, **und** *als Stopwörter konfiguriert haben.*

<map>

Mit dem Tag `map` definieren Sie einen Block, in dem Sie festlegen, wie die Systempfade der gefundenen Dokumente in der Ergebnisliste angezeigt werden. Der Pfad zum jeweiligen Verzeichnis wird dann bei der Verlinkung in der Trefferliste durch den Wert ersetzt, den Sie vorgegeben haben.

Syntax:

```
<map "absoluter/Systempfad">
 to = "[URL]"
</map>
```

Beispiel:

```
<map "/usr/local/share/imperia/htdocs">
 to = "http://ihr-server.net"
</map>
```

Per Default wird das Verzeichnis, das Sie in der Datei system.conf Ihres Systems als `DOCUMENT-ROOT` angegeben haben, durch das `ABS-DOC-ROOT` ersetzt.

Hinweis:

Beim Mapping werden lange Pfadangaben vor kurzen verglichen.

<output>

Zur Steuerung der Ausgabe des Suchergebnisses stehen Ihnen eine Reihe von Direktiven zur Verfügung, mit denen Sie beispielsweise die Anzahl der Ergebnisse festlegen, die pro Bildschirmseite angezeigt werden. Diese Anweisungen notieren Sie in einem Block, den Sie durch das Tag „output“ kennzeichnen.

Syntax:

```
<output>
 [Direktiven]
</output>
```

In der Datei `site/config/fts.conf.sample` Ihrer **imperia** -Installation finden Sie eine Beispielkonfiguration mit den Default-Werten. Die Direktiven für die Steuerung der Ausgabe sind in dem Abschnitt über Output-Direktiven erklärt.

Tags zur Template-Steuerung

Mit den Tags zur Template-Steuerung legen Sie fest, wo sich die Templates auf dem Zielsystem befinden, die von der Volltextsuche verwendet werden. Sie haben die Möglichkeit, innerhalb eines Tags mehrere Alternativen anzugeben, so dass ein schneller Wechsel zwischen unterschiedlichen Layoutvarianten möglich ist. Die Auswahl der Variante geschieht über einen Parameter, den Sie dem Suchskript mitgeben.

Wenn Sie jeweils nur eine Template-Variante benötigen, diese aber durch direkte Links erreichbar machen wollen, müssen sich die Templates unterhalb des DOCUMENT-ROOT des Zielsystems befinden, das durchsucht werden soll. Lesen Sie hierzu auch den Abschnitt zur Programmierung von Suchtemplates im Programmierhandbuch.

Syntax:

```
<[Template-Steuertag]>
 [Zahl] = "/pfad/zum/template"
</[Template-Steuertag]>
```

Die Zahl ist dabei der Schlüssel, über den das betreffende Template beim Aufruf des Suchskripts referenziert wird. Möglich sind absolute Pfadangaben oder gültige, zum Verzeichnis cgi-bin relative Pfade.

<standardHTMLTemplate>

Innerhalb dieses Tags geben Sie die Templates für einfache Suchanfragen an.

Beispiel:

```
<standardHTMLTemplate>
 1 = "/usr/local/share/imperia/htdocs/search/search-form.default.html"
</standardHTMLTemplate>
```

<advancedHTMLTemplate>

Templates für verfeinerte Suchanfragen notieren Sie innerhalb dieses Tags.

Beispiel:

```
<advancedHTMLTemplate>
 1 = "/imperia/site/fts/templates/search-form.default.html"
</advancedHTMLTemplate>
```

<resultHTMLPage>

Notieren Sie hier die Templates für die Darstellung der Suchergebnisse.

Beispiel:

```
<resultHTMLTemplate>
 1 = "/imperia/site/fts/templates/search-form.default.html"
</resultHTMLTemplate>
```

<templateDynamicGroups> und <template>

Wenn Sie für die Indexerstellung Meta-Gruppen definiert haben, können Sie im Suchtemplate eine Select-Box zur Verfügung stellen, die **imperia** automatisch mit den vorhandenen Gruppen befüllt. Das Select-Feld im Suchtemplate müssen Sie hierfür mit dem Namen *GROUP* versehen und diesen in doppelte Anführungszeichen setzen. Mit dem Tag `<templateDynamicGroups>` bestimmen Sie, welche Gruppen mit welchem Namen in welchem Template automatisch als Optionen erscheinen sollen.

Syntax:

```
<templateDynamicGroups>
        <template [Template-Nummer]>
                Meta-Gruppenname = "Text für Option"
        </template>
</templateDynamicGroups>
```

Mit dem Tag `<template>` legen Sie fest, in welchem Suchtemplate die automatische Generierung der Optionen erfolgen soll. Ersetzen Sie den Platzhalter `[Template-Nummer]` durch die Nummer des entsprechenden Templates, die Sie bei den Template-Tags (siehe oben) angegeben haben. Als Meta-Gruppennamen tragen Sie den Namen ein, den Sie in der Indexkonfiguration verwendet haben. Der Text für die Option erscheint dann im Suchtemplate in der Selectbox. Dazu ein Beispiel:

Auszug aus der `fts.conf`:

```
<templateDynamicGroups>
        <template 1>
                newsFb = "Fußballnachrichten"
                newsHb = "Handballnachrichten"
                newsEh = "Eishockeynachrichten"
```

```
                _all_ = "Treffer aus allen Gruppen"
        </template>
</templateDynamicGroups>
```

Eine Besonderheit ist der Gruppenname `_all_`. Bei Auswahl dieser Option durchsucht **imperia** alle vorhandenen Gruppen.

Im entsprechenden Suchtemplate ist dazu folgendes Select-Feld definiert:

```
<select name="GROUP" multiple="multiple">
        <option>Dummy</option>
</select>
```

Beim Aufruf des Suchtemplates über das Suchskript aus der cgi-bin ersetzt **imperia** die Dummy-Option durch die Gruppen aus der Datei `fts.conf`. Im Suchtemplate erscheint also folgender Code:

```
<select name="GROUP" multiple="multiple">
        <option value="newsFB">Fußballnachrichten</option>
        <option value="newsHb">Handballnachrichten</option>
        <option value="newsEh">Eishockeynachrichten</option>
        <option value="_all_">Treffer aus allen Gruppen</option>
</select>
```

165343 Direktiven in Option-Blöcken

Die folgenden Direktiven müssen Sie innerhalb eines Option-Blocks notieren.

verbosityLevel

Mit dieser Direktive steuern Sie den Detailgrad der Informationen, die am Schluss der Suche im Index ausgegeben werden. Der Default-Wert ist 1.

Hinweis:

Derzeit unterscheiden sich die ausgegebenen Meldungen nicht.

dataDir

Mit dieser Direktive legen Sie das Verzeichnis fest, in dem sich der Index befindet.

Wichtig:

Dieser Wert muss in der Konfiguration der Indexerstellung (Datei index.conf) und der Suche (Datei fts.conf) identisch sein.

Syntax:

```
dataDir = "/absoluter/pfad"
```

Das System legt automatisch für jede konfigurierte Domain ein Unterverzeichnis in dem Verzeichnis an, das Sie mit `dataDir` festgelegt haben.

Beispiel:

```
dataDir = "/usr/local/share/imperia/site/fts/index/mandant1"
```

Der Default-Wert ist das Verzeichnis `site/fts/index` Ihrer **imperia** -Installation.

showContext

Die Anzahl der Wörter, die als Kontext zu einem Treffer angezeigt werden, bestimmen Sie mit dieser Direktive. Wenn Sie den Wert für diese Einstellung auf 0 setzen, deaktivieren Sie die Anzeige von Kontextbegriffen in der Trefferliste.

Syntax:

```
showContext = [Zahl]
```

Beispiel:

```
showContext = 100
```

Der Default-Wert dieser Einstellung ist 100.

percentPrecision

Mit dieser Direktiven bestimmen Sie die Anzahl der Nachkommastellen bei der Anzeige des Werts für die Templatevariable `PAGE_PERCENT`. Der Default-Wert ist 0.

phpEnable

Aktivieren Sie diese Direktive, wenn Sie PHP-Code in Ihrem Suchtemplate parsen lassen möchten, bevor es an den Browser geschickt wird.

Syntax:

```
phpEnable = [Wert]
```

Mögliche Werte zur Aktivierung des PHP-Parsings sind:

- yes
- true
- 1

Um das Feature zu deaktivieren, setzen Sie die Direktive auf einen der folgenden Werte:

- no
- false
- 0

Beispiel:

```
phpEnable = "yes"
```

Per Default ist die Auswertung von PHP-Code deaktiviert..

phpPath

Wenn Sie die Auswertung von PHP-Code im Suchtemplate aktiviert haben, müssen Sie mit dieser Direktive den Pfad zum PHP-Interpreter angeben, wenn sich dieser nicht im Umgebungspfad befindet.

Syntax:

```
phpPath = "absoluter/Pfad/zum/Interpreter/Binary"
```

Beispiel:

```
phpPath = "/usr/local/bin/php"
```

Der Default-Wert ist "*php*".

queryMode

Normalerweise ist aus Performancegründen in Suchbegriffen nur eine Wildcard möglich und bei der Verwendung von Wildcards dürfen Suchbegriffe nur aus einem Wort bestehen. Dieses Standard-Verhalten können Sie mit der Direktiven `queryMode` ändern. Außerdem lässt sich noch ein Suchmodus aktivieren, der die Suche nach Metagruppen ohne Angabe eines Suchworts erlaubt. Die folgende Tabelle erklärt die möglichen Einstellungen.

Parameter	Erklärung
`strict`	Innerhalb von Suchbegriffen ist nur eine Wildcard zulässig (Default).
`loose`	Ermöglicht die Verwendung mehrerer Wildcards in mehreren Suchbegriffen.
`extraloose`	Ermöglicht die Verwendung mehrerer Wildcards innerhalb von Suchwörtern.
`partial`	Die Volltextsuche behandelt Suchwörter ohne Wildcards automatisch so, als wären der erste und der letzte Buchstabe eine Wildcard.
`metaonly`	Ermöglicht die Suche nach Metagruppen ohne Angabe von Suchwörtern. Der Suchbegriff kann dann ausschließlich aus Metagruppen-Namen bestehen. Als Ergebnis erhalten Sie alle Dokumente aus den betreffenden Metagruppen.

Tabelle 16.4. Einstellungen für queryMode: verschiedene Suchmodi

minValidCharacters

Diese Direktive gibt an, wie viele Zeichen außer einer Wildcard in einem gültigen Treffer enthalten sein müssen. Der Default-Wert, 0, ermöglicht es auch, lediglich eine Wildcard als Suchbegriff anzugeben.

highlight

Bei aktivierter Kontextanzeige lassen sich die gefundenen Suchwörter in dem angezeigten Text hervorheben. Dabei hebt die Volltextsuche normalerweise das ganze Wort hervor, auch wenn beispielsweise bei einer Suche mit Wildcard nur ein Teil des Wortes tatsächlich im Suchbegriff vorkommt. Um diese Teile gesondert hervorzuheben, setzen Sie die Direktive `highlight` auf den Wert *partial*. Die gesonderte Hervorhebung ist im Standard-Stylesheet für die Volltextsuche in eigenen CSS-Klassen definiert. Wenn Sie ein eigenes Stylesheet nutzen wollen brauchen Sie folgende Klassen für die Formatierung der Hervorhebungen:

- *highlight*
- *highlight_partial*
- *highlight_meta*

additionalUserVars

Mit dieser Direktive können Sie die Liste der CGI-Parameter für automatisch generierte, selbstreferenzierende Aufrufe der Suche erweitern. Diese zusätzlichen Parameter erscheinen dann beispielsweise in den Aufrufen des Suchskripts bei der automatisch erstellten Verlinkung der Trefferseiten.

Syntax:

```
additionalUserVars = "komma-separierte_Liste"
```

Beispiel:

```
additionalUserVars = "var1,var2,var3"
```

groupBool

In der Konfiguration der Indexerstellung können Sie mit den Direktiven metaGroup und dirGroup Gruppen definieren, die bei Suchanfragen als Filterkriterien fungieren, die mit dem Parameter *GROUP* an das Suchskript übergeben werden (Lesen Sie hierzu auch den Abschnitt „Übergabeparameter und Variablen in Suchtemplates“ im Kapitel „Templates für die Volltextsuche“ im Programmierhandbuch).

Mit der Direktive `groupBool` können Sie die boolesche Verknüpfung dieser Filtergruppen steuern. Mit der Einstellung "*AND*" aktivieren Sie eine Filterung über alle Gruppen. Jeder andere Wert bedeutet, dass Ergebnisse lediglich zu einer der vorhandenen Gruppen gehören müssen.

Syntax:

```
groupBool = [WERT]
```

Beispiele:

```
groupBool = "AND"
```

```
groupBool = "and"
```

Dies sind Beispiele für logische UND-Verknüpfungen. Treffer müssen zu allen mit *GROUP* übergebenen Gruppen gehören, um angezeigt zu werden.

```
groupBool = "OR"
```

```
groupBool = "any"
```

Dies sind Beispiele für logische ODER-Verknüpfungen. Treffer müssen nur einer der mit *GROUP* übergebenen Gruppen angehören, um in der Trefferliste zu erscheinen.

Diese Filterung ist nur aktiv, wenn Sie beim Aufruf des Skripts mehr als eine Gruppe als Filterkriterium mit dem Parameter *GROUP* übergeben. Der Default-Wert ist "*AND*".

defaultOperator

Mit dieser Direktive geben Sie an, wie die Volltextsuche mehrere Wörter in Suchbegriffen verknüpft, wenn der Benutzer keine Operatoren eingibt. Der Standardwert ist "*OR*".

resultFilterPlugin

Mit dieser Direktive aktivieren Sie auf Ihrem System vorhandene Ergebnisfilter zur Aufbereitung der Trefferliste vor der Ausgabe. Notieren Sie als Wert den Package-Namen des Filter-Plug-Ins. Ergebnisfilter-Plug-Ins müssen im Verzeichnis `site/modules/core/Dynamic/FTSResultFilter` liegen.

Syntax:

```
resultFilterPlugIn = Plug-In1,Plug-In2,Plug-InN
```

Mehrere Plug-Ins notieren Sie als kommaseparierte Liste ohne Leerzeichen. Dabei legen Sie gleichzeitig die Abarbeitungsreihenfolge der einzelnen Filter fest. Die Volltextsuche wendet den zuerst notierten Filter auch zuerst auf die Ergebnisliste an. Das Ergebnis dieser Filterung filtert sie anschließend mit dem an zweiter Stelle stehenden Filter und so weiter. Anschließend erfolgt die Ausgabe der gefilterten Ergebnisliste.

Beispiele:

imperia beinhaltet drei verschiedene Ergebnisfilter:

DateBonus

Dieses Plug-In ermöglicht die Sortierung der Filter anhand datumsbasierter Bonuskriterien. Die Vergabe datumsbasierter Boni konfigurieren Sie mit einer Reihe weiterer Direktiven. Lesen Sie hierzu Abschnitt 16.5.3.4.7 **DateBonus: Treffer mit datumsbasierten Boni sortieren** auf Seite 211.

UserSort

Dieser Filter legt die Sortierreihenfolge der Trefferliste fest. Per Default erfolgt die Sortierung nach absteigender Relevanz. Alternative Sortiermethoden bestimmen Sie mit dem CGI-Parameter *SORT*, der im Abschnitt **Templates für die imperia -Volltextsuche** im Programmierhandbuch beschrieben ist. Mit der Direktive *sortBy* bestimmen Sie die Default-Sortiermethode. Lesen Sie hierzu Abschnitt 16.5.3.4.6 **UserSort: Sortierung der Treffer bestimmen** auf Seite 210.

Hinweis:

Wie **imperia** *die Relevanz eines Dokuments berechnet, ist in Abschnitt 16.4* **Berechnung der Treffer-Relevanz und Boni** *auf Seite 179 beschrieben.*

XpdfConvert

Dieser Filter ist nützlich, wenn Sie PDF-Dokumente indizieren und auf der Ergebnisseite das Erstellungs- bzw. letzte Änderungsdatum der PDF-Dokumente anzeigen möchten. Das Programm *pdfinfo*, Bestandteil des XPDF-Pakets, liefert beide Datumsangaben als String im englischen Datumsformat. Dabei können abhängig von der zur Erzeugung des PDF-Dokuments verwendeten Software unterschiedliche Variationen auftreten.

Der Filter *XpdfConvert* wandelt ausgehend von einer Reihe von Varianten diese Datumsangaben in eine Angabe im Format *TT.MM.JJJJ hh:mm* um.

16.5.3.4.4 Direktiven in parseroption-Blöcken

Parseroptions sind eine Unterebene der nornalen Options. Ihre Besonderheit ist, dass sie direkt an alle Parser-Plug-Ins weitergereicht werden. Dadurch ist eine Erweiterung der Konfiguration für neu geschriebene Parser-Plug-Ins möglich.

Beispiel:

```
<option>
        dataDir = "/absoluter/pfad"
        <parseroption>
                ignoreMeta = "title|author"
        </parseroption>
</option>
```

ignoreMeta

ignoreMeta ist eine Option, die vom HTML-Parser ausgewertet wird. Mit ihrer Hilfe ist es möglich, den Inhalt von Metadaten aus dem HTML-Kopf zu ignorieren. Dies ist insbesondere wichtig, wenn wie im folgenden Listing ein Metatag "title" und ein HTML-Titel vorhanden ist.

```
<html>
        <head>
                <title>Seitentitel</title>
                <meta name="title" content="Seitentitel">
        </head>
```

Ohne Verwendung von ignoreMeta="title" würde der Titel hier zweimal indexiert und im Suchergebnis zweimal hintereinander ausgegeben.

Der Wert des Feldes wird als regulärer Ausdruck ausgewertet.

Beispiel:

```
ignoreMeta = "title|author"
```

keepphptag

keepphptag ist eine Option, die vom HTML-Parser ausgewertet wird. Ohne diese Option entfernt der HTML-Parser alle Tags, die mit <? ... ?> eingeschlossen sind, bevor der Rest der Datei geparst wird.

Beispiel:

```
<option>
    <parseroption>
        keepphptag = 1
    </parseroption>
</option>
```

process_ssi

Wird die Option process_ssi auf "1" gesetzt, so werden beim Indexieren der Dokumente SSI-Includes mit eingebunden. Standardmäßig ist diese Option abgeschaltet, was dazu führt, dass Teile von Webseiten, die als SSI eingebunden sind, nicht mit indiziert werden.

Beispiel:

```
<option>
    <parseroption>
        parse_ssi = 1
    </parseroption>
</option>
```

165345 Direktiven in Output-Blöcken

CURRENT_PAGE

Mit dieser Direktive legen Sie fest, welche Ergebnisseite zuerst angezeigt wird.

Syntax:

```
CURRENT_PAGE = [Zahl]
```

Beispiel:

```
CURRENT_PAGE = 1
```

Der Default-Wert ist 1.

MAX_PAGE_LIST

Dies gibt die Höchstanzahl der Verweise auf weitere Trefferseiten an, die pro Seite angezeigt werden.

Syntax:

```
MAX_PAGE_LIST = [Zahl]
```

Beispiel:

```
MAX_PAGE_LIST = 100
```

Der Default-Wert ist 100.

MAX_PAGES

Dies gibt die Anzahl maximal generierter Trefferseiten an.

Syntax:

```
MAX_PAGES = [Zahl]
```

Beispiel:

```
MAX_PAGES = 100
```

Der Default-Wert ist *100*.

PER_PAGE

Dies gibt die Anzahl angezeigter Treffer pro Bildschirmseite an.

Syntax:

```
PER_PAGE = [Zahl]
```

Beispiel:

```
PER_PAGE = 10
```

Der Default-Wert ist 10.

16.5.3.6 UserSort: Sortierung der Treffer bestimmen

Der Ergebnisfilter *UserSort* regelt die Sortierung der Trefferliste. Die jeweilige Sortiermethode legen Sie mit dem CGI-Parameter *SORT* fest, der im Abschnitt **Templates für die imperia -Volltextsuche** im Programmierhandbuch beschrieben ist. Die Default-Sortiermethode bestimmen Sie mit der im Folgenden beschriebenen Direktive *sortBy*.

sortBy

Diese Direktive regelt die Default-Sortiermethode der Trefferliste. Die folgende Tabelle zeigt die verfügbaren Sortiermethoden:

Parameter	Erklärung
`arelevance`	aufsteigende Sortierung nach Relevanz
`drelevance`	absteigende Sortierung nach Relevanz (Default)
`amodified`	aufsteigende Sortierung nach Änderungsdatum
`dmodified`	absteigende Sortierung nach Änderungsdatum
`asize`	aufsteigende Sortierung nach Dateigröße
`dsize`	absteigende Sortierung nach Dateigröße
`awords`	aufsteigende Sortierung nach Wortanzahl
`dwords`	absteigende Sortierung nach Wortanzahl

Tabelle 16.5. Sortiermethoden

disableUserSort

Setzen Sie diese Direktive auf den Wert *1*, um etwaige mit der Suchanfrage übergebene Sortieroptionen zu unterdrücken.

16.5.3.47 DateBonus: Treffer mit datumsbasierten Boni sortieren

imperia bietet einen Ergebnisfilter, mit dem Sie Dokumenten anhand ihrer Aktualität einen Relevanzbonus verleihen können. Wenn Sie den Filter aktiviert haben, definieren Sie Maximalbonus, Intervall und Schrittweite der datumsbasierten Boni. Den Maximalbonus bekommt das jeweils aktuellste Dokument aus einer Trefferliste. Dieser Wert verringert sich im von Ihnen definierten Intervall mit abnehmender Aktualität des betreffenden Dokuments um die die ebenfalls von Ihnen vorgegebene Schrittweite.

bonus_date_max

Mit der Direktive `bonus_date_max` definieren Sie den Relevanzbonus für das aktuellste Dokument aus der Trefferliste. Geben Sie einen Wert zwischen 1 und 100 an.

Beispiel:

```
bonus_date_max = 50
```

bonus_date_step

Mit dieser Direktive geben Sie an, um wieviel sich der Bonus jeweils nach Ablauf eines Intervalls verringern soll. Möglich sind Werte zwischen 1 und 100.

```
bonus_date_step = 10
```

bonus_date_period

Diese Direktive gibt das Intervall an, in dem sich der Bonus für die Aktualität eines Dokuments verringert. Geben Sie eine Zahl und optional eine Zeiteinheit an. Für die Zeiteinheiten gibt es eine Kurz- und eine Langschreibweise, die in der folgenden Tabelle aufgeführt sind.

Kurzform	Langform	Beschreibung
mi	*minutes*	Minuten
d	*days*	Tage
w	*weeks*	Wochen
mo	*months*	Monate
y	*years*	Jahre

Tabelle 16.6. Zeiteinheiten für die Direktive `bonus_date_period`

Wenn Sie keine Zeiteinheit angeben, wählt die Volltextsuche automatisch Monate. Aus den drei Parametern Maximalbonus, Schrittweite und Zeitintervall ergibt sich, wie alt ein Dokument maximal sein kann, um einen Aktualitätsbonus zu erhalten. Dazu ein Beispiel:

```
bonus_date_max = 50
bonus_date_step = 10
bonus_date_period = 1 d
```

Mit dieser Konfiguration bekommen Dokumente, die bis zu einem Tag alt sind, einen Bonus von 50, was dem Maximalwert entspricht. Dokumente, die zwischen einem und zwei Tage alt sind, erhalten einen Bonus von 40 (50 - 10). Mit jedem weiteren Tag verringert sich der Bonus jeweils um weitere 10, bis er ganz aufgebraucht ist. Dokumente, die älter als 5 Tage sind, bekommen also gar keinen Bonus mehr.

bonus_date_field

Per Default nutzt die Volltextsuche das Metafeld `__imperia_modified`, das Datum der letzten Bearbeitung im CMS, zur Berechnung des Datumsbonus'. Grundsätzlich können Sie aber jedes Feld als Basis für diese Berechnung verwenden, das eine Datumsangabe in Form eines Unix-Timestamps enthält. Mit der Variablen `bonus_date_field` bestimmen Sie dieses Metafeld. Folgende Optionen stehen Ihnen zur Verfügung:

Parameter	Erklärung
`file_modified`	Datum der letzten Bearbeitung in **imperia** (Default)
`fs_file_modified`	Änderungsdatum auf Dateisystemebene
`METAFELDNAME`	Beliebiges Metafeld, das einen Unix-Timestamp enthält. Definieren Sie in der Indexkonfiguration eine PageMeta-Direktive für dieses Feld, um es indizieren zu lassen.

Tabelle 16.7. Angaben für die Variable bonus_date_field

Wenn Sie ein beliebiges, selbst definiertes Metafeld als Grundlage für die Berechnung verwenden möchten, müssen Sie es durch die Definition einer entsprechenden PageMeta-Direktive in der Indexkonfigurationsdatei in den Index aufnehmen (siehe Abschnitt 16.5.3.3.5 **Tags und Direktiven in Directory-Blöcken** auf Seite 191).

Beachten Sie auch, dass das Änderungsdatum auf Dateisystemebene sich beispielsweise auch bei einem Reparsing der Datei ändert. Das bedeutet, dass nicht immer zwingend eine Änderung am Inhalt der Datei erfolgt, wenn sich das Datei-Änderungsdatum erneuert.

16.5.3.5 Indexerstellung

Die Erstellung des Suchindex stoßen Sie mit dem Skript `fts_index.pl` im Verzeichnis `site/bin` Ihrer **imperia** -Installation an. Rufen Sie das Skript von der Kommandozeile aus auf. Dabei stehen folgende Aufrufparameter zur Verfügung:

Parameter	Bedeutung
-h, --help	Dieser Parameter gibt die Hilfeseite aus.
-i, --install	Die Installation eines zuvor erstellten Index wird ausgeführt. Die zugehörigen Dateien werden dabei aus dem temporären Verzeichnis in das mit der Einstellung dataDir spezifizierte Verzeichnis kopiert.
-b, --build	Löst die Erstellung des Index für die angegebenen Domains aus.
-t, --test	Startet die Prüfung der Index-Konfiguration und gibt eine Liste verfügbarer Komprimierungs-Plug-Ins aus.

Tabelle 16.8. Einstellungen der Direktive *verbosityLevel*

Außerdem können Sie beim Aufruf noch die Domains angeben, die Sie indizieren möchten. Jede Domain, die Sie hierbei aufführen, muss in der Datei `index.conf` im Verzeichnis `site/config` des Systems, das Sie indizieren möchten, wenigstens mit einem Domain-Block definiert sein.

Beispiel:

```
perl fts_index.pl -bi mandant1 mandant2 mandant3
```

Das Index-Skript liest die Konfigurationsdatei ein und erstellt für alle spezifizierten Domains, die definiert sind, einen Index. Wenn Sie beim Aufruf keine Domain angeben, wird die Standard-Domain "*default*" verwendet. Liegt im Verzeichnis `site/config` keine Konfigurationsdatei vor, setzt das Skript für alle Parameter automatisch die Default-Werte ein.

Der so erzeugte Index liegt zunächst in einem temporären Verzeichnis und ist erst verfügbar, wenn Sie das Skript zur Indexerstellung mit der Option *-i* bzw. *--install* ausführen.

Indexerstellung und -installation können Sie auch in einem Arbeitsgang durchführen. Bei umfangreichen Datenbeständen kann die Indizierung erhebliche Zeit beanspruchen. Es ist empfehlenswert, den Index in regelmäßigen Abständen neu zu erstellen. Dies kann zeitgesteuert und automatisiert erfolgen, beispielsweise durch einen Cron-Job.

16.5.3.6 Erstellung von Suchskripten

Wenn eine Suchanfrage gestartet wird, liest das Suchskript die Einstellungen für das Durchsuchen des Index der zu durchsuchenden Domain aus der Datei `fts.conf` im Verzeichnis `site/config` ein. Wenn Sie mehrere Domains konfiguriert haben, brauchen Sie daher für jede Domain ein eigenes Suchskript.

Um ein domainspezifisches Suchskript zu erstellen, genügen wenige Schritte:

1. Erstellen Sie eine Kopie des Skripts `cgi-bin/fts_search.pl`.
2. Öffnen Sie die Datei in einem Editor und suchen Sie die folgende Zeile (Zeile 28):

```
my $domain = 'default';
```

3. Ändern Sie die Zeile wie folgt:

```
my $domain = '[Name_Ihrer_Domain]';
```

Beispiel:

```
my $domain = 'mandant1';
```

Speichern Sie die Datei.

Wichtig:

Das Installationsskript für Hotfixe benennt möglicherweise die im System vorhandene Version der Datei cgi-bin/fts_search.pl um und installiert die Ursprungsversion des Skripts. Daher sollten Sie domainspezifische Anpassungen immer in einer Kopie der Datei vornehmen und diese statt der Originaldatei für den Einsatz der Volltextsuche verwenden.

Bei Suchanfragen für diese Domain verweisen Sie nun auf diese bearbeitete Kopie des Suchskipts.

16.5.3.7 Stopwortlisten anlegen und verwalten

Um eine Stopwortliste für die **imperia** -Volltextsuche einzurichten, sind folgende Schritte notwendig:

1. **Stopwortlisten-Datei erstellen**

Sie benötigen zunächst eine gewöhnliche ASCII-Textdatei mit beliebiger Dateieendung, die Sie an beliebiger Stelle in der **imperia** - Verzeichnisstruktur ablegen.

Tipp

Es ist auch möglich, diese Datei als mit **imperia** *pflegbares Dokument zu verwalten. Beachten Sie in diesem Fall aber, dass die so erzeugte Textdatei im Bereich liegt, der über den Webserver erreichbar ist. Eventuell ist es also notwendig, den Zugriff auf diese Datei einzuschränken, beispiesweise durch einen Passwortschutz.*

In der Datei sind die einzelnen Stopwörter jeweils in einer eigenen Zeile aufzuführen wie das folgende Beispiel zeigt:

```
ist
und
im
in
an
am
bei
beim
ein
```

```
eine
dem
der
die
das
```

Neben den eigentlichen Stopwörtern benötigen Sie in der Datei keinerlei weitere Anweisungen. Einmal angelegte Stopwortlisten können Sie jederzeit ersetzen, ändern oder entfernen.

2. **Stopwortlisten-Eintrag generieren**

   Eine Stopwortlisten-Datei kann die **imperia** -Volltextsuche noch nicht direkt verarbeiten. Um eine Stopwortlisten-Datei für das System verfügbar zu machen, rufen Sie das Skript `site/bin/fts_blacklist.pl` wie folgt auf:

   ```
   perl fts_blacklist.pl -b BEZEICHNER pfad/zur/stopwortliste.end
   ```

   Den mit dem Parameter *-b* spezifizierten *BEZEICHNER* verwenden Sie später in der Konfiguration der Indexerstellung und der Suche zur eigentlichen Aktivierung der Stopwortliste. Pfadangaben ohne führenden Slash interpretiert das System relativ zum Skript, mit führendem Slash absolut. Das Skript erzeugt in einem eigenen Verzeichnis parallel zum Suchindex eine mit dem *BEZEICHNER* benannte Datei, die mit dem Perl-Modul Storable serialisiert und somit für die Volltextsuche verarbeitbar ist.

   Weitere Skriptparameter dienen zur Verwaltung bestehender Stopwortlisten. Die folgende Tabelle listet alle verfügbaren Parameter auf und erklärt sie.

Parameter	Bedeutung
-h, --help	Dieser Parameter gibt die Hilfeseite aus.
-i, --input_charset	Zeichensatz der Stopwortlisten-Datei, Default ist UTF-8.
-b, --blacklist_name	Pflichtparameter, der den Bezeichner der Stopwortliste angibt. Das System speichert die Wörter aus der angegebenen Stopwortlisten-Datei unter diesem Bezeichner, der auch zur Einbindung in der Konfiguration der Indexerstellung und Suche dient.
-a, --append	Das Skript hängt die Wörter aus der angegebenen Datei an die mit *-b BEZEICHNER* spezifizierte Stopwortliste an, statt diese zu ersetzen.
-l, --list	Listet alle im System verfügbaren Stopwortlisten-Einträge auf. Hierfür brauchen Sie weder Bezeichner noch Stopwortlisten-Datei anzugeben.
-d, --delete	Löscht den mit *-b BEZEICHNER* spezifizierten Stopwortlisten-Eintrag aus dem System. Eine Stopwortlisten-Datei brauchen Sie nicht anzugeben. Es ist nicht möglich, auch Stopwortlisten-Dateien mit diesem Parameter zu löschen.

Tabelle 16.9. Parameter für `site/bin/fts_blacklist.pl`

Dazu einige Beispiele:

```
perl fts_blacklist.pl -b mylist ../config/mylist.conf
```

Dieser Aufruf erzeugt aus der unter `site/config` liegenden Stopwortlisten-Datei `mylist.conf` einen Stopwortlisten-Eintrag, der unter dem Bezeichner *mylist* verfügbar ist. Einen etwaigen bereits bestehenden Stopwortlisten-Eintrag ersetzt diese Skriptausführung durch den neuen Eintrag.

```
perl fts_blacklist.pl -a -b mylist /home/wwwrun/imperia/stopwords/extra_words.txt
```

Dieser Aufruf hängt die Wörter aus der unter `/home/wwwrun/imperia/stopwords/` liegenden Stopwortlisten-Datei `extra_words.txt` an den Stopwortlisten-Eintrag an, der unter dem Bezeichner *mylist* verfügbar ist. Sollte noch kein Eintrag mit diesem Namen vorhanden sein, legt **imperia** ihn an.

```
perl fts_blacklist.pl -d -b mylist
```

Dieser Aufruf löscht den Stopwortlisten-Eintrag, der unter dem Bezeichner *mylist* verfügbar ist.

3. **Stopwortlisten in die Konfiguration einbinden**

   Stopwortlisten aktivieren Sie, indem Sie die Stopwortlisten-Einträge in den Konfigurationsdateien für Indexerstellung (`site/config/index.conf`) und Suche (`site/config/fts.conf`) angeben. Hierfür nutzen Sie in beiden Konfigurationsdateien die Direktive `blacklist`.

   Syntaxt:

   ```
   blacklist = BEZEICHNER
   ```

   Der *BEZEICHNER* ist hierbei der Name des Stopwortlisten-Eintrags, den Sie beim Aufruf von `site/bin/fts_blacklist.pl` mit dem Parameter *-b* verwendet haben. Verwenden Sie die Direktive mehrmals, wenn Sie mehrere Stopwortlisten einsetzen wollen. Wenn Sie mehrere Volltextsuchen-Domains definiert haben, sind Stopwortlisten so definierbar, dass sie für alle oder nur für einzelne Domains gelten. Setzen Sie die Direktive `blacklist` innerhalb eines Domainblocks, damit die Stopwortliste nur für diese Domain gilt, und außerhalb der Domainblöcke in der Konfigurationsdatei, damit die Stopwortliste für alle Domains gültig ist.

Wichtig:

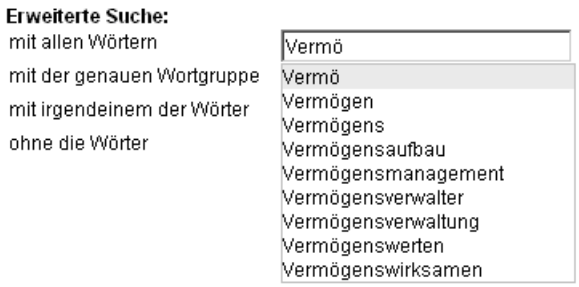
Wenn Sie Stopwortlisten in die Konfiguration der Indexerstellung einbinden, wirken diese sich erst bei der nächsten Indexerstellung aus. Das Gleiche gilt auch bei Änderungen an bereits bestehenden und eingebundenen Stopwortlisten.

Auf Suchanfragen wirken sich Stopwortlisten jedoch sofort nach ihrer Einbindung in die Suchkonfiguration aus. Die Suche ignoriert die Stopwörter sofort unabhängig davon, ob sie noch im Index sind oder nicht.

16.6 Vorschläge in der Volltextsuche

In der **imperia** FTS können nun komfortabel Vorschläge im Suchtemplate angezeigt werden. Wird die Mindestanzahl an Buchstaben eingegeben, üblicherweise 3 oder 4, dann wird per Ajax eine Abfrage an den internen Suchindex erstellt un die ersten n Treffer zur Auswahl vorgschlagen.

Beispiel:



Zum Einrichten dieser Funktion müssen Anpassungen in der Konfiguration des Index (`index.conf`) und der Suchparameter (`fts.conf`) vorgenommen werden. Das Suchtemplate muss erweitert werden und ggf. muss das Ajax-Script für unterzerschiedliche Domains angepasst werden.

16.6.1 Anpassungen an der index.conf

In der Datei `index.conf` muss ein zusätzlicher Index mit den real existierenden Begriffen angelegt werden. Dazu wird im Abschnitt „options“ einer Domain ein neuer Abschnitt „suggest“ angelegt.

Beispiel :

```
<suggest>
   rule            =  "^\S{3,}$" # fallback
</suggest>
```

Die Regel welche Begriffe in den Index gelangen wird duch die Anweisung `rule` gesteuert. Wird der Wert `rule` nicht oder auf den Wert auf `false` gesetzt, so wird kein Index für die Vorschläge angelegt. Somit können keine Vorschläge gemacht werden.

Wird `rule` auf `true` gesetzt, so werden per default alle Worte mit mindestens 3 Zeichen in den Index aufgenommen. Dazu wird intern der reguläre Ausdruck

```
"^\S{3,}$"
```

verwendet. Über die Zuweisung eines andreren regulären Ausdrucks kann der Inhalt des Index gesteuert werden.

```
rule = "^\S{4,}$"
```

16.6.2 Anpassungen an der fts.conf

In der Datei `fts.conf` muss ein Abschnitt „suggest“ angelegt angelgt bzw. bearbeitet werden.

Die grundlegente Konfiguration erfolgt über die folgenden Parameter:

- minChars

  Über den Parameter `minChars` wird die benötigte Anzahl der Zeichen vorgegeben damit die Suche Vorschläge unterbreitet. Üblicherweise werden Vorschläge erst ab 3 oder 4 Zeichen gemacht.

- frequency

  Der Wert `frequency` steuert intern wie oft das Erreichen der `minChars` geprüft werden soll. Der default 1 (s), aber auch kleinere Werte sind möglich.

- maxSuggestions

  Mit dem Wert `maxSuggestions` wird die Anzahl der Vorschläge gesteuert.

- ajaxURI

Der Wert `ajaxURI` muss gesetzt werden und steuert welches Script aus dem Suchtemplate aufgerufen wird. Es existiert kein Defaultwert! CHANGE TO DEFAULT

- formname

  Der `formname` ist der Name des Formulars in dem sich das Suchfeld im Template befinded.

Beispiel :

```
minChars        = 4

frequency       = 0.5

maxSuggestions  = 10

ajaxURI = "/cgi-bin/ajax_fts_suggestions_cms01.pl"

formname = "search"
```

16.6.3 Anpassungen am Template

Jedes Suchfeld im FTS Template kann um die Vorschlags-Funktion erweitert werden. Dazu muss das Formular um die Ajax-Funktionen erweitert werden, wie im folgenden Beispiel:

```
    <form id="search_form" action="/cgi-bin/fts_search.pl" method="post">

    <input id="search-value" type="text" name="search" value="Suchbegriff" style="width:90%" o
    <div id="ifts_suggestions_search-value" class="ifts_suggestions" style="display:none"></di
    <script language="JavaScript" type="text/javascript">
        document.observe('dom:loaded', function () {
            new Ajax.Autocompleter("search-value", "ifts_suggestions_search-value", "/cgi-bin/
                paramName: "suggest",
                parameters: $($('search-value').form.id).serialize(),
                minChars: 3,
                frequency: 0.1,
                afterUpdateElement: function () {$('search-value').form.submit()}
            });
        });
    </script>
        <input type="submit" id="submit" name="submit" value="start" style="width:auto" />&nbs
    </form>
```

In den Resultat-Seiten, die von der Such-Engine ausgewertet werden kann man nach dem Such-Feld z.B. „SEARCH“ das Tag

```
<!--SUGGESTIONS:SEARCH-->
```

hinzufügen. Dadurch wird der oben beschriebene Javascript Teil automatisch eingefügt.

Das oben abgebildete Beispiel kann wie folgt implementiert werden:

```
<div class="searchextended" style="display:none;">
    <h2>Erweiterte Suche:</h2>
    <fieldset>

        <label class="left">mit allen Wörtern</label>
        <input type="text" id="SEARCH_AND"
                              name="SEARCH_AND"
```

```
                            value="[!--SEARCH_REQUESTED--]"
                            size="30" /><br />
        <!--SUGGESTIONS:SEARCH_AND-->

        <label class="left">mit der genauen Wortgruppe</label>
        <input type="text" size="30"
                            id="SEARCH_PHRASE"
                            name="SEARCH_PHRASE" /><br />
        <!--SUGGESTIONS:SEARCH_PHRASE-->

        <label class="left">mit irgendeinem der Wörter</label>
        <input type="text" size="30"
                            id="SEARCH_OR"
                            name="SEARCH_OR" /><br />
        <!--SUGGESTIONS:SEARCH_OR-->

        <label class="left">ohne die Wörter</label>
        <input type="text" size="30"
                            id="SEARCH_NOT"
                            name="SEARCH_NOT" />
        <!--SUGGESTIONS:SEARCH_NOT-->

    </fieldset>
</div>
```

Hinweis:

Obenstehender Code ist aus Darstellungsgründen umgebrochen.

Damit die grundsätzlichen Ajax-Funktionen zur Verfügung stehen, müssen die Java-Script Bibliotheken mit

```
<script language="JavaScript" type="text/javascript"
          src="/imperia/iwl/jscript/dist/prototype.js"></script>
<script language="JavaScript" type="text/javascript"
          src="/imperia/iwl/jscript/dist/effects.js"></script>
<script language="JavaScript" type="text/javascript"
          src="/imperia/iwl/jscript/dist/controls.js"></script
```

Hinweis:

Obenstehender Code ist aus Darstellungsgründen umgebrochen.

in das Template eingebunden werden.

16.6.4 Anpassungen am Ajax-Script

Im Standard wird das Script `ajax_fts_suggestions.pl` mitgeliefert. Wenn Sie nur die „default“ Domain verwenden, müssen Sie hier keine Anpassungen durchführen.

Wenn Sie unterschiedliche Domains verwenden sollten Sie das Script `ajax_fts_suggestions.pl` z.B. in ein Script `ajax_fts_suggestions_MYDOMAIN.pl` für die Domain MYDOMAIN umkopieren und dann die Zeile

```
my $domain = 'default';
```

durch

```
my $domain = 'MYDOMAIN';
```

ersetzen.

Kapitel 17. Variablen zur System-Konfiguration

Imperia verfügt über eine Vielzahl von Möglichkeiten zur Systemkonfiguration. Diese Konfiguration geschieht über Variablen, die in Dateien abgespeichert werden.

Die für den Betrieb wichtigsten Variablen werden während des Installationsprozesses automatisch angelegt. Sie können sie über die Benutzeroberfläche in entsprechenden Dialogfenstern nachträglich ändern, beispielsweise die Einstellungen der Mediendatenbank oder die Konfiguration des Zielsystems.

Zusätzlich zu diesen grundlegenden Variablen gibt es noch eine Reihe weiterer Einstellungsmöglichkeiten, die es Ihnen erlauben, Ihr System an Ihre individuellen Anforderungen anzupassen.

Sie können sämtliche Konfigurationsdateien über die Benutzeroberfläche editieren. Lesen Sie hierzu Abschnitt 2.9 **Konfiguration des Systems über die Benutzeroberfläche** auf Seite 28.

Hinweis:

Wenn **imperia 9** *über ein existentes* **imperia 8** *System installiert wird, wird ein sog. „Compatibilitätslayer" mit installiert. Dieser Compatibilitätslayer interpretiert noch einige der System-Conf Variablen die es in* **imperia 9** *nicht mehr gibt. Somit können unter anderem Projektsrcipte weiter funktionieren. In einem „frischen"* **imperia 9** *System stehen diese Variablen nicht mehr zur Verfügung. Bitte beachten Sie, dass der Compatibilitätslayer nur mit eingeschränkter Funktionalität ausgestattet ist um den Umstieg zu erleichtern.*

Hinweis:

Sowohl der Parametername als auch der Wert müssen in Anführungszeichen gesetzt werden.

Im Folgenden sind die verfügbaren Konfigurations-Variablen thematisch gruppiert aufgeführt.

17.1 Gruppe ACLs

17.1.1 Variable ENABLE_FLEX_ACL

Auf „1" setzen, um ACLs für Flexmodule zu aktivieren.

17.2 Gruppe Anzeige

17.2.1 Variable BOTTOMTABLE

Hinweis:

Nur noch im Compatibilitätslayer!

Geben Sie hier den Pfad zu einer Datei mit einem HTML-Schnipsel an, um die Footer-Tabelle in der Imperia-GUI auszutauschen. Diese Datei muss sich unterhalb des Document-Root des Produktionssystems befinden, z.B: `/my_cool_bottomtable.html`.

17.2.2 Variable CATEGORY_DELETION_PROMPT

Wenn diese Variable den Wert „1" hat, wird vor dem Löschen einer leeren Rubrik ein Bestätigungsdialog angezeigt.

17.2.3 Variable DISABLE-FAV-MENU

Hinweis:

Nur noch im Compatibilitätslayer!

Mit dieser Variablen können Sie das Favoriten-Menü zu- oder abschalten. Setzen Sie sie auf „1“ , um das Menü abzuschalten.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.2.4 Variable DISPLAY_BASE_CLASS

Hinweis:

Nur noch im Compatibilitätslayer!

Ist diese Variable gesetzt, wird für die Darstellung der Imperia-Oberfläche nicht das zur Installation gehörende Modul `site/modules/core/Imperia/Display/Base.pm` benutzt, sondern es kann ein projektspezifisch angepasste Display-Modul genutzt werden.

17.2.5 Variable DONT_ASK_BEFORE_EXPLODE_ALL_NODES

Wenn Sie diese Variable auf „ 1 “ setzen, wird keine Warnmeldung angezeigt, wenn alle Rubriken des Rubrikenbaums angezeigt werden.

17.2.6 Variable FAV_MENU_LINK_ONLY

Hinweis:

Nur noch im Compatibilitätslayer!

Durch diese Variable können Sie die Anzeige von Icons im Favoritenmenü zu- oder abschalten. Der Defaultwert ist 1 für abgeschaltete Icons.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.2.7 Variable IBS_PAGE

Setzen Sie diese Variable auf einen Wert >= 0, um im Archiv eine Paginierung zu aktivieren. Der Wert gibt dabei an, wie viele Dokumente auf einer Seite im Archiv erscheinen sollen. Die Anzeige von Rubriken und Dokumentversionen im Archiv beeinflusst diese Variable nicht.

17.2.8 Variable LATCHING

Mit dieser Variablen schalten Sie die Funktion **Zwischenspeichern** für den Edit-Modus von Dokumenten ein. Im Bearbeiten-Schritt eines Dokuments erscheint dann bei den Steuerelementen ein zusätzlicher Button, mit dem sich eingegebener Inhalt speichern lässt, ohne den Workflow-Schritt zu verlassen.

In der Konfiguration des Workflow-PlugIns Bearbeiten können Sie das Zwischenspeichern auch über den Workflow einstellen. Dabei gibt es drei Möglichkeiten:

- Mit dem Wert `Standard-Einstellung` wird die Einstellung aus der Systemkonfiguration übernommen.
- Mit dem Wert `Verstecken` wird das Zwischenspeichern in diesem Schritt deaktiviert, auch wenn in der Systemkonfiguration etwas andeses steht.
- Mit dem Wert `Zwischenspeichern` wird das Zwischenspeichern in diesem Schritt aktiviert auch wenn in der Systemkonfiguration etwas andeses steht.

17.2.9 Variable MAIN_MENU_LINK_ONLY

Hinweis:

Nur noch im Compatibilitätslayer!

Setzen Sie diese Variable auf *1*, um die Anzeige der Icons im Menü zu unterbinden. Diese Einstellung gilt für alle Benutzer.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.2.10 Variable MD-WINDOW-HEIGHT

Mit dieser Variablen können Sie die Höhe aller Mediendatenbank-Fenster festlegen. Die Angabe erfolgt numerisch und wird als Höhe in Pixeln umgesetzt.

17.2.11 Variable MD-WINDOW-WIDTH

Mit dieser Variablen können Sie die Breite aller Mediendatenbank-Fenster festlegen. Die Angabe erfolgt numerisch und wird als Breite in Pixeln umgesetzt.

17.2.12 Variable POPUP_WINDOW_HEIGHT

Mit dieser Variablen legen Sie die Höhe aller von Imperia geöffenten Pop-Up-Fenster fest. Die Angabe erfolgt numerisch und wird als Höhe in Pixeln umgesetzt.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

17.2.13 Variable POPUP_WINDOW_WIDTH

Mit dieser Variablen legen Sie die Breite aller von Imperia geöffenten Pop-Up-Fenster fest. Die Angabe erfolgt numerisch und wird als Breite in Pixeln umgesetzt.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

17.2.14 Variable TOPTABLE

Hinweis:

Nur noch im Compatibilitätslayer!

Geben Sie hier den Pfad zu einer Datei mit einem HTML-Schnipsel an, um die Header-Tabelle in der Imperia-GUI auszutauschen. Diese Datei muss sich unterhalb des Document-Root des Produktionssystems befinden, z.B: `/my_cool_toptable.html`.

17.3 Gruppe Authentifizierung

17.3.1 Variable AUTH_PLUGIN

Imperia Authentifizierungs-PlugIn.

17.3.2 Variable AUTH_PLUGIN_BACKUP

In dieser Variablen wird der Authentifizierungsmechanismus aus `AUTH_PLUGIN` gespeichert wenn das System in den Maintenancemodus umgeschaltet wird. Beim Zurückschalten wird dann dieser Wert wieder in die Variable `AUTH_PLUGIN` geschrieben.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.4 Gruppe Datenhaltungs-PlugIn Datei

17.4.1 Variable FILE65-DOCUMENT-DIR

Verzeichnis, in dem das File65-PlugIn Dokumente abspeichert.

Der Default-Wert ist *SITE-DIR/documents*.

17.5 Gruppe File Transfer

17.5.1 Variable OPEN_3_USE_SYSREAD

Diese Variable wird auf Linux-Systemen mit neueren Kernelversionen benötigt, wenn Sie Dateitransfer per Secure Copy einsetzen wollen. Setzen Sie sie auf 1, um Sysread bei Secure Copy Transfers zu aktivieren.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.5.2 Variable SSH-OPTIONS

Über diese Variable können Optionen die bei der SSH-Verbindung verwendet werden sollen angegeben werden. Damit sich shell-Verbindungen und Imperia-Verbindungen gleich verhalten, sollten die Oprion besser in der `.ssh` Datei gepflegt werden.

17.5.3 Variable SSH-CLIENT

Der absolute Pfad zur SSH-Binärdatei. Sie benötigen diese Variable, wenn der Default-Wert nicht zutrifft. Der Default-Wert ist *ssh*.

17.5.4 Variable SSH-FILED

In dieser Variablen ist gespeichert, welches Filed-Skript auf Ihrem Zielsystem aufgerufen wird. Der Default ist `SITE-DIR/bin/filed.pl`. Dabei wird SITE-DIR durch die jeweils aktuelle Einstellung auf Ihrem Zielsystem ersetzt.

Der Default-Wert ist *SITE-DIR/bin/filed.pl*.

17.6 Gruppe Flexmodule

17.6.1 Variable ALLOW_PERL_UPLOAD

Mit dieser Variable wird gesteuert, ob Perl-Flexmodule in Imperia bearbeitet werden können.

17.7 Gruppe Grundeinstellungen

17.7.1 Variable CGI-BIN

Hinweis:

Nur noch im Compatibilitätslayer!

Diese Variable enthält den URI des cgi-bin-Verzeichnisses auf dem Webserver, das heißt den Pfad ausgehend vom Document-Root Ihres Servers, beispielsweise '/cgi-bin', '/cgi-shell', '/cgi/perl' usw.

Der Default-Wert ist */cgi-bin*.

17.7.2 Variable CGI-DIR

Hinweis:

Nur noch im Compatibilitätslayer!

In dieser Variablen ist der absolute Pfad zum cgi-bin-Verzeichnis Ihres Webservers gespeichert. Alle CGI-Skripte von Imperia liegen in diesem Verzeichnis.

17.7.3 Variable DEMOMODE

Diese Variable können Sie als Hintertür zum System für Notfälle nutzen. Sollten Sie einmal keinen Zugriff mehr auf Ihr System habe, kommentieren Sie die entsprechende Zeile in der Datei `site/config/system.conf` aus. Sie können sich danach wieder am System anmelden und haben Superuser-Zugriffsrechte (Jeder Benutzer, der sich dann am System anmeldet, hat Superuser-Zugriffsrechte). Anschließend können Sie einen neuen Benutzer anlegen. Entfernen Sie den Kommentar vor der Zeile, um zum Normalbetrieb zurückzukehren.

Der Default-Wert ist *no*.

17.7.4 Variable DISABLE-FLOCK

Setzen Sie diese Variable auf 1, um File Locking zu deaktivieren. Dies ist notwendig, wenn das Betriebssystem, unter dem Sie Imperia installieren, diesen Mechanismus nicht unterstützt (z.B. Windows 98).

17.7.5 Variable DOCTREE_PLUGIN

Verwendetes Datenhaltungs-PlugIn.

Der Default-Wert ist *File65*.

17.7.6 Variable ARCHIVE_DISABLE_DELETE

Setzen Sie diese Variable auf "1", um die Löschfunktion im Archiv zu deaktivieren.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.7.7 Variable FollowSymLinks

Mit dieser Variablen können Sie Teile des Dateisystems einbinden, die außerhalb des Document-Root Ihres Systems liegen. Wenden Sie diese Variable nur in dringenden Fällen an, da Sie hiermit möglicherweise Ihr System für Angriffe verwundbar machen.

17.7.8 Variable IMPERIA-VERSION

Imperia Versionsnummer, bitte nicht ändern.

17.7.9 Variable IMPERIA_ERROR_LOG

Setzen Sie diese Variable auf „1“ für ein eigenes Error-Log. Die Datei finden Sie unter „`SITE-DIR/logs/error.log`“.

Hinweis:

Diese Variable wird in Windows-Installationen automatisch aktiviert.

17.7.10 Variable MAX_REQUEST_SIZE

Mit dieser Variable legen Sie die maximale Größe in Bytes für Dateien beim Upload fest. Die Angabe muss numerisch erfolgen. Versuche, Dateien hochzuladen, die größer sind als dieser Wert, beantwortet das System mit einer Fehlermeldung. Wenn diese Variable nicht gesetzt wird, können Dateien von unbegrenzter Größe hochgeladen werden.

Diese Variable wurde mit Imperia Version 6.5.2 eingeführt.

17.7.11 Variable MIME_TYPES

Mit dieser Variablen können Sie einen alternativen Pfad zu Ihrer MIME-Type-Datenbank definieren. Hier können projektspezifische Erweiterungen updatesicher eingetragen werden; beachten Sie, dass die benötigten Typen aus der Originaldatei mit eingetragen werden müssen, um in Kraft zu bleiben.

17.7.12 Variable SECRET_FREQUENCY

Anzahl der Sekunden, nach deren Verstreichen das Secret-Cookie erneuert wird.

Der Default-Wert ist *3600*.

17.7.13 Variable SECURE_UMASK

Maske zur Generierung des Secure-Verzeichnisses.

17.7.14 Variable DOCUMENT-ROOT

Diese Variable enthält den absoluten Pfad zum Document-Root Ihrer Imperia-Installation. Beispiel: „DOCUMENT-ROOT“ =„/usr/httpd/htdocs“ .

Hinweis:

Das Verzeichnis „imperia“ sollte sich unterhalb des Basis-Verzeichnisses für Webdokumente Ihres Webservers befinden, beispielsweise „/usr/http/htdocs/imperia“ .

17.7.15 Variable OPERATING-SYSTEM

Das Betriebssystem, unter dem Imperia installiert ist. Entweder „UNIX“ oder „DOS“, je nachdem, was dem von Ihnen verwendeten System "am nächsten" kommt.

Der Default-Wert ist *UNIX*.

17.7.16 Variable SITE-DIR

Dies is eine der wenigen Konfigurationsvariblen die nicht in der Datei `/site/config/system.conf` steht sondern in der Datei `/cgi-bin/imperia.conf` die unter anderem dem Auffinden der Datei `/site/config/system.conf` dient.

In dieser Variablen steht der absolute Pfad zum site-Verzeichnis Ihrer Installation.

Wichtig:

Aus Sicherheitsgründen sollten die drei Verzeichnisse SITE-DIR, CGI-BIN und DOCUMENT-ROOT voneinander getrennt sein. Das heißt, dass keines dieser drei Verzeichnisse sich innerhalb eines der anderen befinden sollte. Andernfalls sind unbeabsichtigte oder böswillige Zugriffe durch Benutzer über das Internet möglich.

17.8 Gruppe HTTP Cookie

17.8.1 Variable COOKIE_NAME

Name des Imperia-Cookies.

17.8.2 Variable COOKIE_TTL

Zur Bestimmung der Gültigkeitsdauer (Time To Live) des Imperia Cookies können Sie diese Variable verwenden. Die Angabe erfolgt numerisch und gibt die Lebensdauer des Cookies in Sekunden an. Ist ein Benutzer über den mit dieser Variable festgelegten Zeitraum untätig, wird seine Session automatisch gelöscht und er muss sich erneut anmelden. Der Default-Wert ist 7200 Sekunden.

17.9 Gruppe Hermes

17.9.1 Variable HERMES_RECONNECT

Durch setzen des Schalters `HERMES_RECONNECT` auf „1“ wird bei jedem Hermes durchlauf ein neuer Connect zur Datenbank aufgebaut.

Diese Variable wurde mit Imperia Version 8.5.4 eingeführt.

17.9.2 Variable MAX_AUTO_PUBLISH_DOCS

Diese Variable gibt die Höchstzahl von Dokumenten an, die innerhalb einer Schleife des Hermes-Daemons freigeschaltet werden. Der Default-Wert ist „25“.

17.9.3 Variable HERMES_INCOMING_DELAY

Bei einigen FTP-Servern ist der Lock-Mechanismus während Dateiübertragungen fehlerhaft. Mit dieser Variablen können Sie eine Anzahl von Sekunden angeben, die das MessageCheck-PlugIn mit der Verarbeitung nach der letzten Änderung einer Datei pausiert, um diesem Fehler entgegenzuwirken.

17.10 Gruppe History

17.10.1 Variable DEFAULT_HISTORY_LENGTH

Anzahl der Einträge, die in der History angezeigt werden.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

17.10.2 Variable HISTORY_HIDE_DATE

Wenn Sie diese Variable auf den Wert „ yes “ setzen, bekommen nur Benutzer mit Zugriff auf die Superuser-Rolle in der History Datumseinträge angezeigt.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

17.10.3 Variable HISTORY_HIDE_DATE_SU

Mit dieser Variable können Sie die Datumseinträge in der History vor allen Benutzern verbergen. Benutzer mit Superuser-Zugriff sind eingeschlossen.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

17.11 Gruppe I18N

17.11.1 Variable DEFAULT_LOCALE

Default-Lokalisierungs-Einstellung.

Der Default-Wert ist *C*.

17.11.2 Variable DEFAULT_LANGUAGE

Ländercodes, z.B. DE oder UK.

17.11.3 Variable SYSTEM-LANGUAGE

Die ausgewählte Systemsprache.

17.11.4 Variable SYSTEM_LOCALE

Lokalisierungs-Einstellung des Systems.

17.11.5 Variable USER_LOCALE

Lokalisierungs-Einstellungen der Benutzer.

17.12 Gruppe Imperia Develop Session

17.12.1 Variable ADDR_AUTH

Mit dieser Variablen definieren Sie den Modus, in dem pack_ipaddr arbeitet. Mögliche Werte sind „ PROXY “ oder „ FIX “ .

17.12.2 Variable CRYPT_ALGO

Der Verschlüsselungs-Algorithmus für die Cookies des Session Managements auf dem Produktionsserver.

Hinweis:

Für weitere Informationen siehe Crypt::CBC(3).

Der Default-Wert ist *DES_PP*.

17.13 Gruppe Datum und Zeit

17.13.1 Variable CALENDAR_TIME_FORMAT

Diese Variable dient zur Festlegung des Zeitformats von Datumsangaben, die Sie mit dem Imperia Kalender-Widget generieren. Sie können zwischen 12- und 24-stündigem Zeitformat wählen, indem Sie die Variable auf *12* bzw. *24* setzen.

17.13.2 Variable CALENDAR_DISABLED

Diese Variable dient dazu den Kalender zu deaktivieren.

17.13.3 Variable DATE_COMBO_MINUTES_STEP

Bei der Verwendung von Dropdownfeldern zur Eingabe von Datums- und Zeitangaben kann hier das Raster in Minuten angegeben werden, in dem der Benutzer die Zeit einstellen kann. Beispielsweise kann der Benutzer bei der Angabe "15" nur noch folgende Minutenwerte wählen: 0, 15, 30 und 45.

Ist eine Minutenangabe in einem Dokument bereits erfasst, die nicht ins Raster passt, wird auf den nächstgelegenen Rasterwert gerundet.

Wichtig:

Ist ein Wert für DATE_COMBO_MINUTES_STEP vergeben, können im Javascript-Widget keine Zeiteinstellungen (Stunden und Minuten) mehr gemacht werden.

```
"DATE_COMBO_MINUTES_STEP" = "10"
```

17.13.4 Variable DATE_COMBO_YEARS_LIMIT

Werden Dropdownfelder zur Datumseingabe benutzt, kann man durch Setzen dieser Konfigurationsvariable mit dem Wert „1“ die Auswahlmöglichkeiten im Jahr-Dropdown einschränken. Standardmäßig wird im Jahres-Dropdown eine Liste vom aktuellen Jahr bis 15 Jahre in die Zukunft ausgegeben. Dieser Bereich lässt sich verändern. Bei einer Angabe von „2“ werden bei einem neuen Dokument z.B. nur das aktuelle Jahr und die kommenden beiden Jahre im Dropdown angeboten.

Liegt beim Bearbeiten einer bereits bestehenden Seite das bestehende Datum plus der angegebene Bereich in der Vergangenheit, so wird ganz oben zusätzlich das aktuelle Jahr angeboten.

```
"DATE_COMBO_YEARS_LIMIT" = "2"
```

17.13.5 Variable DISPLAY_TIME_ZONE

Standardmäßig werden Zeit- und Datumsangaben in der Imperia-Oberfläche (also z.B. Freischalt- oder Löschdatum) passend zur Zeitzone angezeigt, die der Nutzer im Betriebssystem eingestellt hat. Ist dies unerwünscht, kann diese Umrechnung mit folgender Angabe in der Datei `site/config/system.conf` deaktiviert werden:

```
"DISPLAY_TIME_ZONE" = "UTC"
```

Beim Start von **imperia** wird erst das Cookie gelesen in dem **imperia** die Zeitzone des Benutzers speichert. Ist dieses Cookie vorhanden, so wird die Zeitzone des Benutzers daraus verwendet. Ist das Cookie nicht vorhanden oder enthält keinen gültigen Wert, so wird die Zeitzone des Systems neu ausgelsen und anschließend in einem Cookie gespeichert. Über die Zeitzone wird dann die Differenzzeit zur Serverzeit bestimmt. Daraus werden dann wiederum die anzuzeigenden Zeiten für den Benutzer berechnet.

Wenn das `AUTH_PLUGIN` nicht „Password“ ist, dann wird in alle Imperia-Seiten das JavaScript `/imperia/jscript/timezone.js` durch das Modul `Imperia::Display::Base.pm` eingebungen.

Die gesetzte Benutzerzeit ist nie relevant, da sie mehr oder weniger varieren kann. Es wird immer die Serverzeit mit dem Offset der im Browser eingestellten Zeitzone verwendet.

Üblicherweise wird die Zeitzone beim Login über den Browser ermittelt und im Benutzerprofil abgespeichert. In seltenen Fällen kann es bei einigen wenigen Browsern dazu kommen, dass die Zeitzone beim Login nicht richtig bestimmt werden kann. In diesen Fällen kann die Zeitzone über die Schaltfläche Zeitzone „zurücksetzen“ erneut gesetzt werden.

17.14 Gruppe Link-Checker

17.14.1 Variable LINK_CHECK

Link-Checks mit HTTP-Requests aktivieren.

17.14.2 Variable NOPROXY

Ohne Proxy.

17.14.3 Variable PROXY_FTP

FTP-Proxy.

17.14.4 Variable PROXY_HTTP

HTTP-Proxy.

17.15 Gruppe Lock Constants

Hinweis:

Die System-Conf Variablen in dieser Gruppe sind nur unter MS Windows relevant!

17.15.1 Variable LOCK_EX

Interner Wert für das File-Locking.

Der Default-Wert ist *2*.

17.15.2 Variable LOCK_NB

Interner Wert für das File-Locking.

Der Default-Wert ist *4*.

17.15.3 Variable LOCK_SH

Interner Wert für das File-Locking.

Der Default-Wert ist *1*.

17.15.4 Variable LOCK_UN

Interner Wert für das File-Locking.

Der Default-Wert ist *8*.

17.16 Gruppe Media Asset Management

17.16.1 DISABLE_8BIT_FILENAMES

Setzen Sie die system.conf-Variable DISABLE_8BIT_FILENAMES auf den Wert "1" um die Umschreibung von 8-bit-Datei- und Verzeichnisnamen in einen "sicheren" Zeichensatz beim MAM-Upload zu erzwingen.

17.16.2 Variable MAM_CAT_ORDER_BY_NAME

Aktivieren Sie die alphabetische Sortierung des Asset-Rubrikenbaums, indem Sie diese Variable auf *1* setzen.

17.16.3 Variable MAM_CATEGORY_TEMPLATE

Geben Sie systemweit den Namen Ihres unter `site/templates` gespeicherten Templates zur Erstellung von Asset-Rubriken an. Die Angabe erfolgt ohne Dateiendung. Fehlt diese Variable, erfolgt das Erstellen von Asset-Rubriken mit dem Standarddialog für die Rubrikenerstellung.

17.16.4 Variable MAM_DISABLE

Setzen Sie diese Variable auf *1*, wenn Sie das Media Asset Management deaktivieren und stattdessen die Imperia-Mediendatenbank verwenden möchten.

Wichtig:

Mit Imperia-Version 8.5 löst das Media Asset Management die Mediendatenbank ab. Diese wird jedoch aus Kompatibilitätsgründen vorübergehend weiter unterstützt.

Das MAM ist dann weiterhin über das Menü verfügbar, und Sie können es weiterhin befüllen und administrieren, aber es erfolgt beispielsweise keinerlei Speicherung von Referenzen über das MAM. Um die Mediendatenbank wieder nutzen zu können, müssen Sie manuell einen Untermenüpunkt erstellen. Lesen Sie hierfür ???.

17.16.5 Variable MAM_ENABLE_PUBLISH

Wenn Sie Ihre Media-Assets unabhängig von Dokumenten freischalten möchten, setzten Sie diese Variable auf *1*.

17.16.6 Variable MAM_GET_UPLOAD_FILENAME

Diese Variable aktiviert ein zusätzliches Eingabefeld in der Upload-Box in einem Asset-Template. Vorbelegt ist dieses Feld mit dem Standard-Dateinamen aus der Asset-Rubrik. Löschen Sie die Vorbelegung und lassen Sie das Feld leer, wenn Sie den ursprünglichen Dateinamen für das Asset übernehmen möchten.

```
"MAM_GET_UPLOAD_FILENAME" = "1"
```

17.16.7 Variable MAM_HIDDEN_FIELDS

Diese Variable beeinflusst in der Listenansicht die Anzeige der Asset-Details und der Spalten sowie die entsprechenden Links in der Iconbox-Ansicht. Notieren Sie eine kommaseparierte Liste der Spalten, deren Anzeige Sie unterbinden möchten. Das System wertet die Liste als regulären Ausdruck aus. Dazu ein Beispiel:

```
"MAM_HIDDEN_FIELDS" = "directory, .*filename" => m/^directory|.*filename$/"
"MAM_HIDDEN_FIELDS" = "directory,template" => m/^directory|template$/"
```

Die folgende Tabelle listet die verfügbaren Spaltennamen auf:

Deutscher Spaltenname (Listen-Ansicht)	FeldID
- (Checkbox)	__mam_select
Zoom	__mam_zoom
Icon	__mam_icon
Titel	__mam_title
Dateiname	__mam_filename
Verzeichnis	-
Dateigröße	__mam_filesize
Details	__mam_details
Referenzen	__mam_references
Löschen	__mam_delete
Erzeugt von	__mam_createdby
Erzeugungsdatum	__mam_createdat
Template	-
Abmessungen	__mam_dimensions
Benutzeraktion	__mam_action
Benutzervarianten	__mam_variants
Archiv	__mam_archive
Import	__mam_reimport

Tabelle 17.1. Verfügbare Felder im Media Asset Management

Hinweis:

Das Feld **Icon** *lässt sich nur in der Listen-Ansicht verbergen. Für Benutzer mit Superuser-Rolle hat das Ausblenden keinen Effekt.*

17.16.8 MAM_JPEG_QUALITY

Mit diesem Parameter können Sie einstellen, in welcher Qualität das MAM JPEG-Varianten von Bildern erstellt. Der Parameter beeinflusst die sichtbare Qualität und die Dateigröße. Grundeinstellung ist 85; der Wertebereich von 20 bis 100. Höhere Werte führen zu besserem Aussehen, erhöhen aber auch die Dateigröße. Mit kleineren Werten kann man an der Dateigröße sparen, muss aber Qualitätseinbußen in Kauf nehmen.

```
"MAM_JPEG_QUALITY" = "75"
```

17.16.9 Variable MAM_LINKS_STYLE

Mit der Variablen `MAM_LINKS_STYLE` legen Sie die Darstellung der Links in den Spalten der Listen-Ansicht fest. Die Syntax lautet wie folgt:

```
"MAM_LINKS_STYLE" = "[Option]"
```

Diese drei Optionen stehen zur Auswahl:

ICON
: In den Spalten erscheinen lediglich kleine Icon-Grafiken.

TEXT
: In den Spalten erscheinen ausschließlich Textlinks.

ICON,TEXT
: In den Spalten erscheinen sowohl Icons als auch Text. Diese Einstellung ist der Default.

17.16.10 Variable MAM_MULTIUPLOAD_OPTIONS

Mit der Variablen MAM_MULTIUPLOAD_OPTIONS richten Sie die verfügbaren Optionen für den Mehrfachupload ein. Die Angabe der Optionen erfolgt in Form einer kommaseparierten Liste von Zahlen.

Beispiel:

```
"MAM_MULTIUPLOAD_OPTIONS" = "2,3,4,5,6,7,8,9,10"
```

Hinweis:

Ab einer Anzahl von ca. 15 Assets ist mit steigenden spürbaren Performance-Einbußen zu rechnen. Das ist aber nur ein Richtwert. Je nach System-Gegebenheiten kann es dazu auch schon früher kommen.

17.16.11 Variable MAM_MULTIUPLOAD_SIZE

Die Option **Datei**->**Mehrfachupload** erlaubt das Anlegen mehrerer Assets in einem Arbeitsgang. Mit der Variablen `MAM_MULTIUPLOAD_SIZE` legen Sie die Standardanzahl hierfür fest.

Beispiel:

```
"MAM_MULTIUPLOAD_SIZE" = "5"
```

17.16.12 Variable MAM_NO_EXTENDER_NORMALIZE

Beim Hochladen von Mediendateien werden Endungen von erkannten Medienformaten in die in der Datei `site/config/mime.types` angegebene Endung umgeschrieben. So wird beispielsweise die Datei "beispiel.jpg" im Standardfall als "beispiel.jpeg" in dem MAM abgelegt. Durch Setzen der Variablen MAM_NO_EXTENDER_NORMALIZE auf den Wert "1" in der `site/config/system.conf` kann das Ändern der Anhänge verhindert werden.

Bitte beachten Sie auch den Abschnitt 17.7.11 **Variable MIME_TYPES** auf Seite 223.

Diese Variable wurde mit Imperia Version 8.5.4 eingeführt.

17.16.13 Variable MAM_PUBLIC_SPOOL

Ist diese Variable auf "1" gesetzt, dürfen alle Benutzer über das Skript `/cgi-bin/site_mam_spool_import.pl` Assets in alle MAM-Rubriken importieren. Informationen zum Skript finden Sie im Abschnitt 11.2 **Import von Assets aus einem Ordner oder einer Zip-Datei** auf Seite 142.

17.16.14 Variable MAM_READ_ONLY_FIELDS

Mit dieser Variablen beeinflussen Sie das Details-Popup der Listenansicht. Legen Sie fest, welche Felder nicht vom Benutzer editierbar sein sollen. Per Default sind dies die Felder `directory`, `filename` und `template`.

17.16.15 Variable MAM_SAVED_FILTERS

Mit dieser Variablen können Sie gespeicherte Filter für alle Benutzer zur Verfügung stellen. Die Beschreibung dazu finden Sie in Abschnitt 11.3.1 **Globale Suchfilter definieren** auf Seite 146.

17.16.16 MAM_SPOOL_CHARSET

Ist Ihr Betriebssystem-Zeichensatz nicht UTF-8, so muss dieser bei der Verwendung des MAM-Massenimports angegeben werden, um eine korrekte Übernahme von Sonderzeichen in die Imperia-Oberfläche und Logfiles zu erzielen.

17.16.17 Variable MAM_SPOOL_PATH

Mit dieser Variablen können Sie steuern aus welchem Verzeichnis Zip Dateien ins MAM mit dem Script `/cgi-bin/mam_spool_import.pl` (siehe Abschnitt 11.2 **Import von Assets aus einem Ordner oder einer Zip-Datei** auf Seite 142) hochgeladen werden können.

```
"MAM_SPOOL_PATH" = "/my_spool"
```

17.16.18 Variable MAM_UNIQUE

Setzen Sie diese Variable auf *1*, wenn Sie möchten, das Imperia alle Assets sowie die zugehörigen Varianten im Dateisystem in einem Unterverzeichnis der betreffenden Rubrik abspeichert. So können Sie beispielsweise verhindern, dass sich beim Verschieben von Assets zufällig identisch benannte Asset-Dateien oder Varianten überschreiben.

17.17 Gruppe Auslieferung

17.17.1 Variable DYNAMIC_URI_PREFIX

Legen Sie systemweit fest, wie das Präfix lauten soll. Default ist `/imp`.

17.18 Gruppe One-Click-Edit

17.18.1 Variable ONECLICKEDIT

Legen Sie systemweit fest, wie One-Click-Edit etwaige Kopien eines Dokuments behandeln soll. Der Defaultwert, *DOCUMENT*, stellt OCE auf dokumentbasierte Verarbeitung ein. Setzen Sie die Variable auf *FILE*, wenn Sie beispielsweise einzelne Kopien eines Dokuments löschen möchten.

Hinweis:

Wenn Sie das Verhalten der OCE-Toolbar auf Dokumentenebene variabel einstellen möchten, empfiehlt sich die Festlegung über das Template. Lesen Sie hierfür den Abschnitt **One-Click-Edit** *in der Syntaxreferenz im Kapitel* **Templates** *des Programmierhandbuchs.*

17.19 Gruppe PHP

17.19.1 Variable PHP_CLI

Typ der verwendeten PHP Version. Setzen Sie diese Variable auf einen wahren Wert, wenn Sie die CLI-Variante des PHP-Interpreters im Einsatz haben. Per Default geht das System von der CGI-Variante aus.

17.19.2 Variable PHP_PATH

Pfad zum PHP-Interpreter.

17.20 Gruppe Publisher

17.20.1 Variable DEFAULT_VERBOSITY

Für ausführlichere Ausgaben beim Freischalten setzen Sie diese Variable auf „ 1 “.

17.20.2 Variable INFORM_ROLE_ON_PUBLISH

Wenn Sie möchten, dass beim Freischalten in Imperia Benachrichtigungen verschickt werden, setzen Sie diese Variable auf „ 1 “ .

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

17.20.3 Variable SERIALIZER

Die Serialisierungsmethode, die der Publisher nutzt. Mit „DUMPER“ wählen Sie das Modul Data::Dumper(3), ohne diese Variable Storable. Bei jedem anderen Wert sucht das System nach einem PlugIn dieses Namens im Verzeichnis `site/modules/core/Dynamic/Serialize` Ihrer Imperia-Installation.

Achtung:

Änderungen an dieser Variablen können zu Datenverlusten führen.

Der Default-Wert ist *Storable*.

17.21 Gruppe Registrierungsschlüssel

Hinweis:

Diese Variablen sollten nicht per „Hand“ gesetzt werden. Beim Eintragen des System-Schlüssels werden diese Werte ausgefüllt.

17.21.1 Variable REG_DEMO

Demo-Registrierungsschlüssel.

17.21.2 Variable REG_KEY

Ihr Imperia Registrierungsschlüssel.

17.21.3 Variable REG_NAME

Ihr Imperia Registrierungsname.

17.22 Gruppe Reparsen

17.22.1 Variable REPARSE_ACTION_CHECKED

Wählen Sie eins der folgenden Ziele für Dokumente, die reparst werden: „live“, „workflow“ und „pending“. Diese Einstellung kann von Benutzern geändert werden. Per Default ist die Option aktiviert.

17.22.2 Variable REPARSE_ACTION_FIXED

Wählen Sie eins der folgenden Ziele für Dokumente, die reparst werden: „live“, „workflow“ und „pending“. Diese Einstellung kann von Benutzern nicht geändert werden.

17.22.3 Variable REPARSE_SOURCE_CHECKED

Mit dieser Variablen können Sie bestimmen, welche Quellfelder vorausgewählt sind. Zur Auswahl stehen „live“ und „pending“ (bei Mehrfachauswahl durch Komma trennen). Mit einer leeren Zeichenkette „“ legen Sie fest, dass es keine vorausgewählten Werte gibt. Diese Einstellung kann von den Benutzern geändert werden. Per Default ist die Option aktiviert.

17.22.4 Variable REPARSE_SOURCE_FIXED

Mit dieser Variablen können Sie die Quelle auswählen. Zur Auswahl stehen „live“ und / oder „pending“ (bei Mehrfachauswahl durch Komma trennen). Mit einer leeren Zeichenkette „“ legen Sie fest, dass es keine vorausgewählten Werte gibt. Diese Einstellung kann nicht von Benutzern geändert werden.

17.23 Gruppe Rubriken

17.23.1 Variable CATEGORY_TEMPLATE

Template zur Bearbeitung der Metainformationen Ihrer Rubriken.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

Der Default-Wert ist *Fallback-Template*.

17.23.2 Variable DEFAULT_DIRECTORY

Standard-Verzeichnis für Rubriken.

Der Default-Wert ist /.

17.23.3 Variable DEFAULT_FILENAME

Standard-Dateiname für Dokumente.

Der Default-Wert ist *index.html*.

17.24 Gruppe Schreibtisch

17.24.1 Variable DESKTOP_COLUMN_MODULES

Eine Liste der Module für Schreibtischspalten. Die einzelnen Elemente werden durch Semikola getrennt. Module müssen in einem Verzeichnis unterhalb von site/modules/core abgelegt sein.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.24.2 Variable DESKTOP_CONFIG_MODULES

Eine Liste der Module zur Schreibtisch-Konfiguration. Die einzelnen Elemente werden durch Semikola getrennt. Module müssen in einem Verzeichnis unterhalb von site/modules/core abgelegt sein.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.25 Gruppe SiteActive

17.25.1 Variable SAVEPAGE-PASSWORD

Geben Sie hier ein Passwort ein, um „ SAVEPAGE “ in site_active.pl über CGI nutzen zu können.

17.25.2 Variable SCHEDULE

Doppelpunktseparierte Liste zusätzlicher Schedule-Dateien.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

17.26 Gruppe Zielsysteme

Hinweis:

Nur noch im Compatibilitätslayer!

17.26.1 Variable live-url

Die absolute URL des Servers, z.B. `http://localhost.at.my.cool.server:8000/`.

17.26.2 Variable liveftp1-directory

Der absolute Pfad zur Document Root auf dem Ziel-Server, z.B. /usr/local/share/imperia/htdocs.

17.26.3 Variable liveftp1-ip

IP-Adresse und Port des Standard-Zielsystems. Diese Variable ist nur für die Übertragungsart FTP relevant.

17.26.4 Variable liveftp1-name

Diese Variable enthält den Benutzernamen für den Verbindungsaufbau zum Standard-Zielsystem mittels FTP .

17.26.5 Variable liveftp1-pass

Diese Variable speichert das Passwort für den Verbindungsaufbau zum Standard-Zielsystem mittels FTP.

17.26.6 Variable liveftp1-site

In dieser Variablen ist der absolute Pfad zum site-Verzeichnis Ihrer Installation gespeichert, z.B. /usr/local/share/imperia/site.

17.26.7 Variable liveftp1_passive_mode

Wenn diese Variable auf „ on “ gesetzt ist, wird der passive FTP-Übertragungsmodus genutzt.

17.26.8 Variable transfer

In dieser Variable ist der aktive Dateiübertragungsmodus gespeichert. Der Default-Wert ist „ Filecopy “.

17.27 Gruppe Template

17.27.1 Variable ALLOW_TEMPLATE_FORM

Setzen Sie diese Variable auf „ 1 “ , um Formulare in HTML-Templates zuzulassen.

17.27.2 Variable ALLOW_TEMPLATE_FRAME

Setzen Sie diese Variable auf „ 1 “ , um Frames in HTML-Templates zuzulassen.

17.27.3 Variable ALLOW_TEMPLATE_SCRIPT

Setzen Sie diese Variable auf „ 1 “ , um Scripting in HTML-Templates zuzulassen.

17.27.4 Variable MD-DEFAULT-LINKS

Mit dieser Variablen legen Sie fest, welche Links im Mediendatenbankaufruf angezeigt werden. Mögliche Werte sind „ none “ , „ upload, mam “ oder „ mam, upload “ .

Der Default-Wert ist *mam, upload*.

17.27.5 Variable SAFE_TEMPLATE

Setzen Sie diese Variable auf „ 1 “ , damit externer Perl-Code (beispielsweise aus Flex-Modulen) in einer Safe(3)-Umgebung ausgeführt wird.

17.27.6 Variable TEMPLATE_EWE

Setzen Sie diese Variable auf „ 1 “ , um die Verarbeitung des EWE-Wordmoduls im Template-Prozessor zu aktivieren.

17.28 Gruppe Templateprozessor

17.28.1 Variable TEMPLATE_CHAIN

Beschreibt systemweit die Reihenfolge der vom Template-Prozessor verwendeten Module. Entfernen Sie nicht benötigte Module aus der Kette, um die Performance Ihres Systems zu steigern.

Achtung:

Die Entfernung zwingend notwendiger Module oder Änderungen der Standard-Reihenfolge können zu unerwarteten Ergebnissen bei der Verarbeitung der Templates führen.

Wenn Sie die in Imperia implementierten Funktionen für mehrsprachige Dokumente nutzen möchten, müssen Sie eine modifizierte Template-Chain definieren. Setzen Sie das Modul **Multilang** nach dem ersten "CodeInclude" und vor "Flex" in die Prozesskette und das Modul **MultilangFinish** an deren Ende.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

Hinweis:

Die Metavariable `__imperia_template_chain` *ist das rubrikweit gültige Äquivalent dieser Konfigurationsvariable.*

17.29 Gruppe Benutzer-Management-Einstellungen

17.29.1 Variable DATABASE_STORAGE_PLUGIN

Datenhaltungs-PlugIn für das Benutzer Management. Entweder „ FileStore “ oder „ LDAP “ .

Der Default-Wert ist *FileStore*.

17.29.2 Variable SORT_ROLES

Setzen Sie diese Variable auf „ IDs “ , um die Rollen nach ihren IDs sortieren zu lassen.

17.29.3 Variable UMLIGHT_PLUGIN

Benutzer-PlugIn.

Der Default-Wert ist *FilestoreLight*.

17.30 Gruppe Workflow

17.30.1 Variable CONFIRM_NEXT_STEP

Wenn Sie diese Variable auf „ 1 “ setzen, muss jeder Workflow-Schritt bestätigt werden.

17.30.2 Variable APPROVAL_DISABLE_JUMP

Wenn Sie diese Variable auf „ 1 “ setzen, können Dokumente, die sich im Genehmigungs-Schritt befinden, nicht mehr in vorhergehende Workflowschritte versetzt werden.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

Kapitel 18. Die imperia Admin-Skripte

Die **imperia** -Admin-Skripte sind als Werkzeuge für Systemadministratoren gedacht. Sie können damit häufig auszuführende Aufgaben vereinfachen. Zur Ausführung rufen Sie das betreffende Skript von der Kommandozeile aus auf. Sie liegen im Verzeichnis site/bin Ihrer **imperia** -Installation. Ein Zugriff über die **imperia** -Benutzeroberfläche ist nicht möglich.

Wichtig:

Legen Sie vor der Verwendung der Admin Skripte ein Backup an bzw. stellen Sie sicher, dass ein aktuelles Backup existiert.

Versetzen Sie **imperia** *außerdem in den Wartungsmodus mit Superuserzugriff (siehe Abschnitt 2.2.2* **Wartungsmodus mit Superuser-Zugriff** *auf Seite 6) bzw. stellen Sie sicher, dass kein anderer Benutzer über die Weboberfläche arbeitet.*

Die Skripte liegen alle im Verzeichnis *site/bin*. Eine Erklärung zu ihrer Anwendung und zu den jeweils verfügbaren Optionen erhalten Sie mit dem Aufrufparameter *--help* . Zum Beispiel:

```
admin_accesscat.pl --help
```

18.1 admin_accesscat.pl

Mit diesem Skript können Sie die Zugriffsrechte für eine Rubrik anzeigen lassen und ändern.

18.2 admin_accessmeta.pl

Legen Sie die Zugriffsrechte für ein Metafile fest oder lesen Sie die gegenwärtigen Zugriffsberechtigungen.

18.3 admin_accesstemplate.pl

Mit diesem Skript ändern Sie die Zugriffsrechte für ein Template oder lesen die aktuelle Konfiguration aus.

18.4 admin_accessworkflow.pl

Mit diesem Skript ändern oder lesen Sie die Zugriffsrechte für einen Workflow.

18.5 admin_acl_update.pl

Mit diesem Skript löschen Sie alte, nicht mehr gültige Rollen aus den ACLs.

18.6 admin_addcat.pl

Mit diesem Skript legen Sie eine neue Rubrik oder Unterrubrik an.

Die Eigenschaften der neuen Rubrik definieren Sie mit Hilfe der Aufrufparameter. Dabei sind alle Angaben optional. Einzige Ausnahme ist der Rubrikname, den Sie angeben müssen.

Bei Unterrubriken müssen Sie zusätzlich den eindeutigen Namen oder die ID der übergeordneten Rubrik angeben.

Hinweis:

Nach der Verwendung von admin_addcat.pl sollten Sie die Rubrikenrechte mit dem Skript admin_accesscat.pl einrichten.

18.7 admin_adddoc.pl

Mit diesem Skript erstellen Sie ein neues Dokument, dessen Attribute Sie über die Aufrufparameter festlegen. Per Default wird anschließend der Workflow für das neue Dokument angestoßen. Dies können Sie jedoch auch durch einen Paramter unterbinden.

18.8 admin_addrole.pl

Mit diesem Skript legen Sie eine neue Rolle an. Den Namen der neuen Rolle und einen Kommentar können Sie beim Aufruf des Skriptes übergeben.

Mit *admin_deriverole.pl* (siehe auch Abschnitt 18.14 **admin_deriverole.pl** auf Seite 238) können Sie die neue Rolle anschließend einer anderen Rolle als Unterrolle zuteilen bzw. der neuen Rolle eine andere Rolle als Unterrolle zuweisen.

Nutzen Sie das Skript *admin_grpuser.pl* (siehe Abschnitt 18.21 **admin_grpuser.pl** auf Seite 238), um Benutzern Zugriff auf diese Rolle zu gewähren.

Für die Festlegung der Berechtigungen der neuen Rolle stehen Ihnen die Admin-Skripte *admin_accesscat.pl* (Rubrikenrechte, siehe auch Abschnitt 18.1 **admin_accesscat.pl** auf Seite 236), *admin_accesstemplate.pl* (Templates, siehe auch Abschnitt 18.3 **admin_accesstemplate.pl** auf Seite 236), *admin_accessmeta.pl* (Metafiles, siehe auch Abschnitt 18.2 **admin_accessmeta.pl** auf Seite 236) und *admin_accessworkflow.pl* (Workflows, siehe auch Abschnitt 18.4 **admin_accessworkflow.pl** auf Seite 236) zur Verfügung.

18.9 admin_adduser.pl

Nutzen Sie dieses Skript, um einen neuen Benutzer anzulegen. Die Attribute des neuen Benutzers können Sie beim Aufruf des Skriptes übergeben.

Zugriffsrechte erteilen Sie dem neuen Benutzer, indem Sie mit dem Skript *admin_grpuser.pl* seine Rollenzugehörigkeiten festlegen.

Sie können anderen Benutzern mit dem Skript *admin_edituser.pl* Zugriff auf den neuen Benutzer gewähren.

18.10 admin_cleandesktop.pl

Dieses Skript findet und löscht (shreddert) alle MAM Dokumente im Schreibtisch, die keine assozierten Binärdaten enthalten. Dokumente dieser Art entstehen beim Abbruch des Uploads.

18.11 admin_currentfile.pl

Mit diesem Skript können Sie Ihre Site nach Dateien durchsuchen, die weder im longlive.log noch in current.files aufgeführt sind. Wahlweise können Sie sich eine Liste dieser Dateien anzeigen lassen oder diese direkt in den Datenbestand des CMS integrieren.

18.12 admin_delrole.pl

Mit diesem Skript löschen Sie eine Rolle aus dem System. Als Parameter können Sie die ID oder den eindeutigen Namen der Rolle angeben.

18.13 admin_deluser.pl

Mit diesem Skript löschen Sie einen Benutzer aus dem System. Als Parameter können Sie die ID oder den eindeutigen Login-Namen des Benutzers angeben.

18.14 admin_deriverole.pl

Nutzen Sie dieses Skript, um sich die bestehende Rollenhierarchie anzeigen zu lassen oder Veränderungen daran vorzunehmen. Sie können damit eine bestehende Rolle einer oder mehreren anderen als Unterrolle zuweisen.

18.15 admin_expire.pl

Mit diesem Skript können Sie Dokumente, die ein Verfallsdatum eingetragen haben, wahlweise anzeigen oder in die Datei `site/articles/expiry.new` schreiben lassen.

18.16 admin_getcat.pl

Mit diesem Skript können Sie die Eigenschaften einer Rubrik ausgeben. Es stehen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können. Die Rubrik legen Sie durch Angabe der NodeID (beginnend mit einem Slash) oder des eindeutigen Rubriknamens fest.

18.17 admin_getdoc.pl

Sie können sich mit diesem Skript die Metainformation zu einem Dokument ausgeben lassen. Dabei stehen Ihnen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können. Das Dokument bestimmen Sie durch die Angabe der NodeID beim Aufruf des Skripts.

Wenn Sie die Informationen einer archivierten Version des Dokumentes anzeigen lassen möchten, geben Sie zusätzlich die arch_id der betreffenden Version an. Andernfalls werden die Informationen der neuesten Version ausgegeben.

18.18 admin_getrole.pl

Mit diesem Skript können Sie die Eigenschaften einer Rolle anzeigen lassen. Die Rolle bestimmen Sie durch die Angabe der RollenID oder des eindeutigen Rollennamens. Es stehen Ihnen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können.

18.19 admin_getuser.pl

Mit diesem Skript können Sie die Eigenschaften eines Benutzers anzeigen lassen. Den Benutzer bestimmen Sie durch die Angabe der UserID oder des eindeutigen Login-Namens des Benutzers. Es stehen Ihnen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können.

18.20 admin_grpaccess.pl

Mit diesem Skript können Sie die Zugriffsberechtigungen auf die Rollen Ihres Systems anzeigen lassen oder ändern. Entziehen oder gewähren Sie Benutzern Zugriff auf eine Rolle.

18.21 admin_grpuser.pl

Mit diesem Skript können Sie sich anzeigen lassen, welche Rollen Ihres Systems ein Benutzer annehmen darf oder einem Benutzer Zugriff auf weitere Rollen gewähren.

18.22 admin_listcat.pl

Nutzen Sie dieses Skript, um sich eine Liste aller Rubriken unterhalb einer Startrubrik anzeigen zu lassen.

18.23 admin_listflex.pl

Rufen Sie dieses Skript auf, um sich eine Liste aller Flexmodule inklusive ihrer Validcodes, Invalidcodes und Autoren anzeigen zu lassen.

18.24 admin_listmeta.pl

Für eine Liste aller Metafiles Ihres Systems rufen Sie dieses Skript auf.

18.25 admin_listrole.pl

Mit diesem Skript können Sie sich eine Liste aller Rollen ausgeben lassen.

18.26 admin_listtemplate.pl

Mit diesem Skript können Sie eine Liste aller Templates erzeugen und ausgeben lassen.

18.27 admin_listuser.pl

Nutzen Sie dieses Skript, um sich eine Liste aller Benutzer anzeigen zu lassen.

18.28 admin_listworkflow.pl

Mit diesem Skript können Sie eine Liste aller Workflows anzeigen lassen.

18.29 admin_sanitisearchive.pl

Nutzen Sie dieses Skript für die Bereinigung des gesamten Archives oder einzelner Teilbereiche.

Geben Sie beim Aufruf eine NodeID oder einen eindeutigen Rubrikennamen an, um nur einen Teilbereich des Archivs zu bearbeiten. Sollen auch Unterrubriken bereinigt werden, müssen Sie dies beim Aufruf des Skripts vermerken.

Alternativ können Sie das ganze Archiv bereinigen lassen.

Für die Bereinigung stehen Ihnen zwei Optionen zur Auswahl. Sie können wahlweise nur sämtliche unbeendeten Dokument-Versionen löschen, die älter sind als die jeweils letzte beendete, oder auch zusätzlich die Anzahl verbleibender beendeter Versionen einschränken.

18.30 admin_searchdoc.pl

Mit diesem Skript suchen Sie Dokumente nach bestimmten Suchkriterien.

Sie erhalten eine Liste von Dokumenten, auf die Ihre Suchkriterien zutreffen. Neben der NodeID können Sie sich eine frei definierbare Reihe von Metafeldinhalten der gefundenen Dokumente anzeigen lassen.

Zur Eingrenzung der Suche können Sie beim Aufruf die NodeID oder den eindeutigen Namen einer Startrubrik übergeben.

18.31 admin_searchuser.pl

Mit diesem Skript können Sie anhand bestimmter Suchkriterien nach Benutzern suchen.

Sie erhalten eine Liste aller Benutzer, auf die diese Kriterien zutreffen. Per Default enthält diese Liste die UID und den Login-Namen der gefundenen Benutzer. Weitere Felder können Sie durch Parameter beim Aufruf der Liste hinzufügen.

18.32 admin_setcat.pl

Mithilfe dieses Skripts können Sie die Eigenschaften bestehender Rubriken ändern.

Die Rubrik bestimmen Sie durch die Angabe einer NodeID oder eines eindeutigen Namens beim Aufruf.

Die Werte, die Sie ändern möchten, übergeben Sie in Form einer durch Leerzeichen getrennten Reihe von Name-Wertpaaren.

18.33 admin_setcounter.pl

Nutzen Sie dieses Skript, um die internen Counter von **imperia** zu verändern. Sie haben Zugriff auf die Counter für Rubriken und Dokumente sowie für Verzeichnisse.

18.34 admin_setdoc.pl

Mit diesem Skript können Sie die Metainformationen eines Dokumentes verändern. Das Dokument wählen Sie durch die Angabe einer NodeID beim Aufruf aus.

Per Default werden die Informationen der letzten Version des Dokumentes verändert. Optional können Sie auch die der letzten beendeten Version oder die einer per archID bestimmten Version aus dem Archiv verändern.

18.35 admin_setrole.pl

Für die Änderung von Rolleneigenschaften können Sie dieses Skript nutzen.

Übergeben Sie dem Skript die ID oder den eindeutigen Namen der Rolle, die Sie ändern möchten, und eine durch Leerzeichen getrennte Reihe von Name-Wertpaaren mit den neuen Eigenschaften.

18.36 admin_setuser.pl

Mit diesem Skript können Sie die Eigenschaften eines Benutzers ändern.

Übergeben Sie dem Skript die ID oder den eindeutigen Namen des Benutzers, den Sie ändern möchten, und eine durch Leerzeichen getrennte Reihe von Name-Wertpaaren mit den neuen Eigenschaften.

18.37 admin_statdoc.pl

Mit diesem Skript können Sie Informationen zum Status eines Dokuments erhalten.

Index

Symbole

.meta, 171
.mnu, 127
.simple, 171
/imperia/documentation/de/imperia, 61
<advancedHTMLTemplate>, 203
<bonus>, 184
<directory>, 188
<directoryMatch>, 188
<domain>, 183, 201
<exclude>, 185
<files>, 189
<groupname>, 198
<include>, 185
<lang>, 201
<map>, 202
<meta>, 198
<metaGroup>, 187
<option>, 187, 201
<output>, 202
<pageMeta>, 191
<pageMetaComment>, 186
<resultHTMLPage>, 203
<settype>, 183
<standardHTMLTemplate>, 203
<templateDynamicGroups>, 203
[delete], 13
[edit], 13
__imperia_category, 171
__imperia_children
    oce_clone, 25
__imperia_clobber_by_id, 172
__imperia_clobber_by_name, 172
__imperia_created
    oce_clone, 24
__imperia_imported, 170
__imperia_meta_expand
    oce_clone, 24
__imperia_modified
    oce_clone, 24
__imperia_node_id
    oce_clone, 24
    oce_delete, 25
    oce_edit, 24
    oce_erase, 25
    oce_history, 25
    oce_info, 26
    oce_linkcheck, 26
    oce_reparse, 26
    oce_unpublish, 26
__imperia_parent
    oce_clone, 25
__imperia_quick_edit
    oce_edit, 24
    oce_linkcheck, 26
__imperia_show_in_cat_filter, 72
__imperia_uid
    oce_clone, 24

A

aberkennen
    Zugriffsrecht(e), 72
abgleichen
    Systemdateien, 60
Ablehnen bei ..., 50, 52
abschalten
    Benutzer-Zugriff, 5
ACL-Verwaltung, 43
additionalUserVars, 206
ADIR, 85
Admin-Skripte
    Archiv bereinigen, 239
    Auflistung aller Metafiles, 239
    Ausgabe der Metainformationen eines Dokuments, 240
    Ausgabe einer Liste aller Rollen, 239
    Ausgabe einer Liste von Flexmodulen, 239
    Benutzer löschen, 237
    Benutzereigenschaften ändern, 240
    Benutzern Rollen zuweisen, 238
    Benutzern Rollenzugriff entziehen, 238
    Benutzern Rollenzugriff gewähren, 238
    Counter ändern, 240
    Dokumente der Datei expiry.new hinzufügen, 238
    Eigenschaften einer Rolle ausgeben, 238
    Eigenschaften eines Benutzers ausgeben, 238
    Liste aller Benutzer anzeigen, 239
    Liste aller Templates anzeigen, 239
    Liste aller Workflows erstellen, 239
    Metainfo eines Dokuments ausgeben, 238
    nach Benutzern suchen, 239
    nach Dokumenten suchen, 239
    neue Rolle anlegen, 237
    neue Rubrik anlegen, 236
    neuen User anlegen, 237
    neues Dokument anlegen, 237
    Nicht mehr gültige Rollen aus ACLs löschen, 236
    Rolle löschen, 237
    Rolleneigenschaften ändern, 240
    Rollenhierarchie ändern, 238
    Rollenhierarchie anzeigen, 238
    Rollenzugriffsberechtigungen anzeigen, 238
    Rollenzugriffsberechtugungen eines Benutzers anzeigen, 238
    Rubrikeigenschaften ändern, 240
    Rubrikeneigenschaften ausgeben, 238
    Rubrikenrechte ändern, 236
    Rubrikenrechte anzeigen, 236
    Schreibtisch aufräumen, 237
    Status eines Dokumentes anzeigen, 240
    suchen nach Dateien, die nicht in longlive.log oder current.files enthalten sind, 237
    Unterrubriken anzeigen, 238
    verfallene Dokumente anzeigen, 238
    Zugriffsrechte für ein Metafile festlegen, 236
    Zugriffsrechte für ein Metafile lesen, 236
    Zugriffsrechte für ein Template festlegen, 236
    Zugriffsrechte für ein Template lesen, 236

Zugriffsrechte für einen Workflow festlegen, 236
Zugriffsrechte für einen Workflow lesen, 236
admin_accesscat.pl, 236
admin_accessmeta.pl, 236
admin_accesstemplate.pl, 236
admin_accessworkflow.pl, 236
admin_acl_update.pl, 236
admin_addcatr.pl, 236
admin_adddoc.pl, 237
admin_addrole.pl, 237
admin_adduser.pl, 237
admin_cleandesktop.pl, 237
admin_currentfile.pl, 237
admin_delrole.pl, 237
admin_deluser.pl, 237
admin_deriverole.pl, 238
admin_expire.pl, 238
admin_getcat.pl, 238
admin_getdoc.pl, 238
admin_getrole.pl, 238
admin_getuser.pl, 238
admin_grpaccess.pl, 238
admin_grpuser.pl, 238
admin_listcat.pl, 238
admin_listflex.pl, 239
admin_listmeta.pl, 239
admin_listrole.pl, 239
admin_listtemplate.pl, 239
admin_listuser.pl, 239
admin_listworkflow.pl, 239
admin_sanitisearchive.pl, 239
admin_searchdoc.pl, 239
admin_searchuser.pl, 239
admin_seetdoc.pl, 240
admin_setcat.pl, 240
admin_setcounter.pl, 240
admin_setrole.pl, 240
admin_setuser.pl, 240
admin_statdoc.pl, 240
Ajax, 218
ajax_fts_suggestions.pl, 218
ajaxURI, 216
Alarm-Verzeichnis, 12
Allgemeine Einstellungen, 26
Anbindung
Verzeichnisdienst, 78
ändern
Rubrikreihenfolge, 44
Systemdienste, 13
Änderungen an Workflows, 88
Änderungsverfolgung, 36
CSS-Formate, 36
anlegen
Benutzer, 75
Freischalt-Trigger, 9
Metadatei, 45
Rolle, 73
Rubrik, 40
Systemdienst, 11
Unterrubrik, 40
Workflow, 89
Workflow-Schritt, 92
Anlegen von Rubriken im Media Asset Management, 140
Anmeldekennung
Zielsystem, 8
Anmelden
erstmaliges, 5
Anmeldung
LDAP, 78
Annotate, 98
anzeigen
Menüstruktur, 129
SQL-Lösch-Statements, 17
SQL-Statements, 17
Anzeigen bei .., 50, 52
Arbeitsblatt
bewegen, 94
Architekturmuster, 130
Archiv, 135
Archiv-Bereiniger, 98
ASCII, 170
ASCII-Import, 172
Asset hochladen, 122
Asset Rubrikenparameter
__imperia_show_in_cat_filter, 72
Asset-Aktionen, 96
Asset-Rubriken
allgemeine Informationen, 140
Rubriken-Eigenschaften, 69
Asset-Variante-PlugIn, 103
Asset-Varianten automatisch erzeugen, 103
Assets importieren, 142
Aufbau
Import-Daten, 172
Aufbau von Workflow-Schritten, 88
Aufbau von Workflows, 88
Ausgabe von Meldungen bei der Indexerstellung, 197
Auswahl
Dokumentenhaltung, 15
Auswahlkriterium, 77
AUTH_PLUGIN, 5
Superuser, 6

B

Basissystem, 1, 135
Basisverzeichnis, 12
Baum, 130
bearbeiten
Benutzer, 76
Benutzerdaten, 76
Codeinclude, 57
Flexmodul, 51
Freischalt-Trigger, 10
Menü, 128
Metadatei, 47
Rolle, 75
Rollendaten, 75
Rubrik, 41
Slotmodul, 53

Template, 49
Workflow, 90, 95
Bearbeiten, 99
Bearbeiten +, 101
Bearbeiten von Rubriken im Media Asset Management, 142
Bearbeitungsansicht der Konfigurationsdateien, 28
Bedienung
Grid, 91
Bedingung, 111
OCE, 23
Benutzer, 73, 77
anlegen, 75
bearbeiten, 76
löschen, 77
Benutzer-Account
LDAP, 78
Benutzer-Zugriff
abschalten, 5
Benutzerbedingung
Trigger, 9
Benutzerdaten
bearbeiten, 76
Benutzereinstellungen, 75-76
Benutzersprache, 61
Beschreibung, 126
Rubriken-Informationen, 70
bestimmen
Metadatei, 71
Rubriken-Eigenschaften, 69
Template, 71
bewegen
Arbeitsblatt, 94
Bibliothek, 218
blacklist, 186, 200
Blocksize
Dokumentenhaltung, 16
bonus_date_field, 211
bonus_date_max, 211
bonus_date_period, 211
bonus_date_step, 211
Button-Leiste, 91

C

Cache-Größe für Index bestimmen, 195
Cache-Verzeichnis der Indizierung, 195
cacheDir, 195
cacheSize, 195
CATDOC, 186
CATPPT, 187
chmap, 194
Classpath
Java-VM, 27
Codeinclude
bearbeiten, 57
hochladen, 55
löschen, 57
neu erstellen, 54
Codeinclude-Dateien, 54
Codeinclude-Management, 54
öffnen, 54
Codeinclude-Verzeichnis
erstellen, 54
löschen, 57
Codeincludes, 54
Compatibilitätslayer, 219
compress, 195
content_maxlevel, 174
content_startlevel, 174
Controler, 130
Controller, 130
/system/mailer, 39
copies
oce_delete, 25
CSS-Formate
Änderungsverfolgung, 36
CURRENT_PAGE, 209

D

Daemon, 166
UNIX®, 166
Windows®, 166
Windows®-Systemdienst, 166
Darstellung, 158
data, 170
dataDir, 196, 204
dataTempDir, 196
Dateien beim Indizieren einschließen, 185
Dateien vom Indizieren ausschließen, 185
Dateiliste, 45, 48, 61, 89
Dateiname, 173
Rubriken-Informationen, 70
Dateinamen als Index-Kriterium, 189
Dateirechte, 60
Dateitypen zur Indizierung festlegen, 183
Datenbank
Einschränkungen, 14
Unterschieden, 14
Datenbank(en)
Dokumentenhaltung, 14
Datenbank-Benutzer
Dokumentenhaltung, 15
Datenbank-Hostname
Dokumentenhaltung, 15
Datenbank-Name
Dokumentenhaltung, 15
Datenbank-Server-Port
Dokumentenhaltung, 15
DBI, 17
DBI-Tracing-Dateiname
Dokumentenhaltung, 16
DBI-Tracing-Level
Dokumentenhaltung, 16
Debugging-TTY
Dokumentenhaltung, 16
del_develop
oce_delete, 25
Depublikationsdatum, 110
Depublizierung, 108
Description, 50, 52
Developsystem, 1

Dienstnummer, 12
directory
  oce_delete, 25
  oce_unpublish, 26
dirGroup, 192
disableUserSort, 210
Document Root, 8
documentDir, 196
Dokument
  importieren, 132
  löschen, 133
  reparsen, 132
Dokument-ID, 171
Dokumentation, 61
Dokumentenauswahl
  Dialogfenster-Optionen, 106
  Konfiguration, 106
Dokumentenauswahl-PlugIn, 105
Dokumentenfortschritt, 96
Dokumentenhaltung
  Auswahl, 15
  benutzerdefinierte Schlüssel, 15
  Blocksize, 16
  Datenbank(en), 14
  Datenbank-Benutzer, 15
  Datenbank-Hostname, 15
  Datenbank-Name, 15
  Datenbank-Server-Port, 15
  DBI-Tracing-Dateiname, 16
  DBI-Tracing-Level, 16
  Debugging-TTY, 16
  File65, 14
  Konfiguration, 14
  LongReadLen, 16
  LongTruncOk, 16
  MySQL, 14
  ODBC_MSSQL, 14
  Oracle, 14
  Passwort, 15
  PostgreSQL, 14
  SID, 16
  Statements, 16
  Treiber-Optionen, 16
  Typ "TEXT", 16
Dokumentverzeichnis für Indizierung bestimmen, 196
Domain, 216, 218
Domänen in Suchanfragen, 201
duplizieren
  Workflow-Schritt, 93
Durchsuchen ..., 46, 49, 51, 53
dynamic.conf, 58
Dynamische Module, 58

E

editieren
  Codeinclude, 57
Editor
  Konfigurationsdateien, 28
Eigenschaften
  Workflow-Schritt, 92
Einbindung von Stopwortlisten in die Indexkonfiguration, 186
Einbindung von Stopwortlisten in die Suchkonfiguration, 200
eingeben
  Rubriken-Informationen, 70
Einschränkungen
  LDAP, 78
einstellen
  Zugriffsrechte, 94
Email-Adresse, 77
entfernen
  Zugriffsrecht, 95
Epoche, 24, 170
Ergebnistemplate, 203
Erstellung eines neuen Codeinclude-Verzeichnisses, 54
Erstellung eines neuen Codeincludes, 54
erstmaliges
  Anmelden, 5
erzeugen
  Flexmodul, 50
  Slotmodul, 52
  Tabellen, 17, 21
Eskalation-Ende, 109
Eskalation-Start, 108
Executable
  Java-VM, 27
Existenzprüfung, 109

F

Feldwert, 77
festlegen
  Workflow, 70
Festlegung von Namen für Meta-Gruppen, 198
FF-Variablen, 12
File65, 14
file_maxlevel, 174
file_startlevel, 174
Filecopy, 7
fileMetaKey, 190
filename
  oce_delete, 25
  oce_unpublish, 26
Filter, 159
finished
  oce_edit, 24
  oce_linkcheck, 26
  oce_reparse, 26
Flexmodul, 50
  bearbeiten, 51
  erzeugen, 50
  hochladen, 51
  löschen, 51
Flexmodul-Verwaltung, 50
follow Symlink, 197
followSubDir, 197
force
  oce_erase, 25
formname, 217
Freischalt-Trigger, 8

anlegen, 9
bearbeiten, 10
Darstellung, 10
löschen, 10
Server, 9
Trigger, 8
Zieldatei, 8
Zielformat, 8
freischalten, 2
Freischalten
Trigger, 8
Freischaltliste, 136
Freischalttrigger, 136
frequency, 216
FTP
passiv, 8
FTP-Transfer, 7
FTS, 215
fts.conf, 200, 216

G

Genehmigung, 110
Genehmigung +, 112
gewählte Rollen, 72
Gewichtung der Volltextsuche beeinflussen, 184
gidnumber, 82
GLOBAL_METAKEYS, 132
Globale Suchfilter im Media Asset Management, 146
Globales Metainfo-Objekt, 132-133
Grid, 90
Bedienung, 91
Button-Leiste, 91
PlugIn-Leiste, 91
groupBool, 207
Gruppierungs-Schema, 82

H

Hauptrubrik, 40
Hermes, 2, 166
-b, 168
-d, 168
-e, 168
-h, 168
-i, 168
-k, 168
-p, 168
-s, 168
-t, 168
-u, 168
-z, 168
hermes.pid, 169
Import, 170
Log-Datei, 169
Optionen, 167
Plug-In-Tabelle, 169
Standard-Konfiguration, 170
UNIX®, 166
Windows®, 166
Windows®-Systemdienst, 166
Hermes-Import, 170
__imperia_imported, 170
ASCII, 170
data, 170
Dokumente ersetzen, 171
mask, 170
Metadateien, 170
Plug-In, 170
XML, 170
Zielrubrik, 171
hermes.pid, 169
highlight, 206
Hilfe
OCE, 23
Hintergrund-Daemon, 166
UNIX®, 166
Windows®, 166
Windows®-Systemdienst, 166
Hintergrund-Prozess, 166
hinzufügen
OCE-Skript, 23
Untermenü, 126
hochladen
Codeinclude, 55
Flexmodul, 51
Metadatei, 46
Slotmodul, 53
Template, 48
Horizontalnavigation, 173
HTML-Konverter-PlugIn, 102
htms, 50, 52
HTTP-Transfer, 7

I

IBlock, 173
ignoreMeta, 208
imperia -Account
LDAP, 78
imperia -OCE-Skripte, 24
imperia.schedule, 11, 60
Import
Hermes, 170
Import-Daten
Aufbau, 172
import.conf, 170
index, 193
index.conf, 182, 216
Indexeinstellungen vererben, 194
Indizieren anhand von Metafeldern, 190
Indzierung aktivieren, 193
Informationen
LDAP-Zugang, 79
Inhalte konvertieren, 102
Inhaltsprüfung-PlugIn, 102
inherit, 194
Installationsvoraussetzungen
Volltextsuche, 180
Intervall
Systemdienst, 11
IP-Adresse
Zielsystem, 8
iWE, 62

Konfiguration, 62
AutoDetectLanguage, 67
BaseHref, 64
BodyClass, 65
BodyId, 65
CustomStyles, 65
DefaultLanguage, 67
DocType, 64
FillEmptyBlocks, 67
FontColors, 66
FontFormat, 66
FontNames, 66
FontSizes, 66
MaxUndoLevels, 64
Plugin.add, 67
StartupFocus, 65
ToolbarSets, 67
Konfigurationsdatei, 62
Konfigurationsoptionen, 63

J

Java-PlugIn, 90
Java-Script, 218
Java-VM, 27
Executable, 27
Java-Vm
Classpath, 27

K

keepphptag, 209
Kerberos, 78
Kommentar, 74
Kompressions-Plug-In für Suche auswählen, 195
Konfiguration, 27-28, 216
Dokumentenhaltung, 14
iWE, 62
Konfiguration der Indexerstellung, 182
Konfiguration der Suchanfragen, 200
Konfiguration des Media Asset Managements, 146
Konfigurations-Variablen
ADDR_AUTH, 225
ALLOW_PERL_UPLOAD, 222
ALLOW_TEMPLATE_FORM, 234
ALLOW_TEMPLATE_FRAME, 234
ALLOW_TEMPLATE_SCRIPT, 234
APPROVAL_DISABLE_JUMP, 235
ARCHIVE_DISABLE_DELETE, 223
AUTH_PLUGIN, 221
AUTH_PLUGIN_BACKUP, 221
BOTTOMTABLE, 219
CALENDAR_DISABLED, 226
CALENDAR_TIME_FORMAT, 226
CATEGORY_DELETION_PROMPT, 219
CATEGORY_TEMPLATE, 233
CGI-BIN, 222
CGI-DIR, 222
CONFIRM_NEXT_STEP, 235
COOKIE_NAME, 224
COOKIE_TTL, 224
CRYPT_ALGO, 225
DATABASE_STORAGE_PLUGIN, 235
DATE_COMBO_MINUTES_STEP, 226
DATE_COMBO_YEARS_LIMIT, 226
DEFAULT_DIRECTORY, 233
DEFAULT_FILENAME, 233
DEFAULT_HISTORY_LENGTH, 225
DEFAULT_LANGUAGE, 225
DEFAULT_LOCALE, 225
DEFAULT_VERBOSITY, 231
DEMOMODE, 223
DESKTOP_COLUMN_MODULES, 233
DESKTOP_CONFIG_MODULES, 233
DISABLE-FAV-MENU, 220
DISABLE-FLOCK, 223
DISABLE_8BIT_FILENAMES, 228
DISPLAY_BASE_CLASS, 220
DISPLAY_TIME_ZONE, 226
DOCTREE_PLUGIN, 223
DOCUMENT-ROOT, 224
D O N T _ A S K _ B E F O R E _ E X P L O - DE_ALL_NODES, 220
DYNAMIC_URI_PREFIX, 231
ENABLE_FLEX_ACL, 219
FAV_MENU_LINK_ONLY, 220
FILE65-DOCUMENT-DIR, 221
FollowSymLinks, 223
Gruppe ACLs
ENABLE_FLEX_ACL, 219
Gruppe Anzeige
BOTTOMTABLE, 219
CATEGORY_DELETION_PROMPT, 219
DISABLE-FAV-MENU, 220
DISPLAY_BASE_CLASS, 220
D O N T _ A S K _ B E F O R E _ E X P L O - DE_ALL_NODES, 220
FAV_MENU_LINK_ONLY, 220
IBS_PAGE, 220
LATCHING, 220
MAIN_MENU_LINK_ONLY, 220
MD-WINDOW-HEIGHT, 221
MD-WINDOW-WIDTH, 221
POPUP_WINDOW_HEIGHT, 221
POPUP_WINDOW_WIDTH, 221
TOPTABLE, 221
Gruppe Auslieferung
DYNAMIC_URI_PREFIX, 231
Gruppe Authentifizierung
AUTH_PLUGIN, 221
AUTH_PLUGIN_BACKUP, 221
Gruppe Benutzer-Management-Einstellungen
DATABASE_STORAGE_PLUGIN, 235
SORT_ROLES, 235
UMLIGHT_PLUGIN, 235
Gruppe Datenhaltungs-PlugIn Datei
FILE65-DOCUMENT-DIR, 221
Gruppe Datum und Zeit
CALENDAR_DISABLED, 226
CALENDAR_TIME_FORMAT, 226
DATE_COMBO_MINUTES_STEP, 226
DATE_COMBO_YEARS_LIMIT, 226
DISPLAY_TIME_ZONE, 226

Gruppe Exporter
OPEN_3_USE_SYSREAD, 222
SSH-CLIENT, 222
SSH-FILED, 222
SSH-OPTIONS, 222
Gruppe Flexmodule
ALLOW_PERL_UPLOAD, 222
Gruppe Grundeinstellungen
ARCHIVE_DISABLE_DELETE, 223
CGI-BIN, 222
CGI-DIR, 222
DEMOMODE, 223
DISABLE-FLOCK, 223
DOCTREE_PLUGIN, 223
DOCUMENT-ROOT, 224
FollowSymLinks, 223
IMPERIA-VERSION, 223
IMPERIA_ERROR_LOG, 223
MAX_REQUEST_SIZE, 223
MIME_TYPES, 223
OPERATING-SYSTEM, 224
SECRET_FREQUENCY, 223
SECURE_UMASK, 224
SITE-DIR, 224
Gruppe Hermes
HERMES_INCOMING_DELAY, 225
HERMES_RECONNECT, 224
MAX_AUTO_PUBLISH_DOCS, 224
Gruppe History
DEFAULT_HISTORY_LENGTH, 225
HISTORY_HIDE_DATE, 225
HISTORY_HIDE_DATE_SU, 225
Gruppe HTTP Cookie
COOKIE_NAME, 224
COOKIE_TTL, 224
Gruppe I18N
DEFAULT_LANGUAGE, 225
DEFAULT_LOCALE, 225
SYSTEM-LANGUAGE, 225
SYSTEM_LOCALE, 225
USER_LOCALE, 225
Gruppe Imperia Develop Session
ADDR_AUTH, 225
CRYPT_ALGO, 225
Gruppe Link-Checker
LINK_CHECK, 227
NOPROXY, 227
PROXY_FTP, 227
PROXY_HTTP, 227
Gruppe Lock Constants
LOCK_EX, 227
LOCK_NB, 227
LOCK_SH, 227
LOCK_UN, 227
Gruppe Media Asset Management
MAM_CAT_ORDER_BY_NAME, 228
MAM_CATEGORY_TEMPLATE, 228
MAM_DISABLE, 228
MAM_ENABLE_PUBLISH, 228
MAM_GET_UPLOAD_FILENAME, 228
MAM_HIDDEN_FIELDS, 228
MAM_JPEG_QUALITY, 229
MAM_LINKS_STYLE, 229
MAM_MULTIUPLOAD_OPTIONS, 230
MAM_MULTIUPLOAD_SIZE, 230
MAM_NO_EXTENDER_NORMALIZE, 230
MAM_PUBLIC_SPOOL, 230
MAM_READ_ONLY_FIELDS, 230
MAM_SAVED_FILTERS, 230
MAM_SPOOL_PATH, 231
MAM_UNIQUE, 231
Gruppe Mediendatenbank
DISABLE_8BIT_FILENAMES, 228
MAM_SPOOL_CHARSET, 230
Gruppe One-Click-Edit
ONECLICKEDIT, 231
Gruppe PHP
PHP_CLI, 231
PHP_PATH, 231
Gruppe Publisher
DEFAULT_VERBOSITY, 231
INFORM_ROLE_ON_PUBLISH, 231
SERIALIZER, 232
Gruppe Registrierungsschlüssel
REG_DEMO, 232
REG_KEY, 232
REG_NAME, 232
Gruppe Reparsen
REPARSE_ACTION_CHECKED, 232
REPARSE_ACTION_FIXED, 232
REPARSE_SOURCE_CHECKED, 232
REPARSE_SOURCE_FIXED, 232
Gruppe Rubriken
CATEGORY_TEMPLATE, 233
DEFAULT_DIRECTORY, 233
DEFAULT_FILENAME, 233
Gruppe Schreibtisch
DESKTOP_COLUMN_MODULES, 233
DESKTOP_CONFIG_MODULES, 233
Gruppe SiteActive
SAVEPAGE-PASSWORD, 233
SCHEDULE, 233
Gruppe Template
ALLOW_TEMPLATE_FORM, 234
ALLOW_TEMPLATE_FRAME, 234
ALLOW_TEMPLATE_SCRIPT, 234
MD-DEFAULT-LINKS, 234
SAFE_TEMPLATE, 234
TEMPLATE_EWE, 234
Gruppe Templateprozessor
TEMPLATE_CHAIN, 234
Gruppe Workflow
APPROVAL_DISABLE_JUMP, 235
CONFIRM_NEXT_STEP, 235
Gruppe Zielsystem
live-directory, 233
live-ip, 234
live-name, 234
live-pass, 234
live-url, 233
liveftp1-site, 234
liveftp1_passive_mode, 234

transfer, 234
HERMES_INCOMING_DELAY, 225
HERMES_RECONNECT, 224
HISTORY_HIDE_DATE, 225
HISTORY_HIDE_DATE_SU, 225
IBS_PAGE, 220
IMPERIA-VERSION, 223
IMPERIA_ERROR_LOG, 223
INFORM_ROLE_ON_PUBLISH, 231
LATCHING, 220
LINK_CHECK, 227
live-url, 233
liveftp1-directory, 233
liveftp1-ip, 234
liveftp1-name, 234
liveftp1-pass, 234
liveftp1-site, 234
liveftp1_passive_mode, 234
LOCK_EX, 227
LOCK_NB, 227
LOCK_SH, 227
LOCK_UN, 227
MAIN_MENU_LINK_ONLY, 220
MAM_CAT_ORDER_BY_NAME, 228
MAM_CATEGORY_TEMPLATE, 228
MAM_DISABLE, 228
MAM_ENABLE_PUBLISH, 228
MAM_GET_UPLOAD_FILENAME, 228
MAM_HIDDEN_FIELDS, 228
MAM_JPEG_QUALITY, 229
MAM_LINKS_STYLE, 229
MAM_MULTIUPLOAD_OPTIONS, 230
MAM_MULTIUPLOAD_SIZE, 230
MAM_NO_EXTENDER_NORMALIZE, 230
MAM_PUBLIC_SPOOL, 230
MAM_READ_ONLY_FIELDS, 230
MAM_SAVED_FILTERS, 230
MAM_SPOOL_CHARSET, 230
MAM_SPOOL_PATH, 231
MAM_UNIQUE, 231
MAX_AUTO_PUBLISH_DOCS, 224
MAX_REQUEST_SIZE, 223
MD-DEFAULT-LINKS, 234
MD-WINDOW-HEIGHT, 221
MD-WINDOW-WIDTH, 221
MIME_TYPES, 223
NOPROXY, 227
ONECLICKEDIT, 231
OPEN_3_USE_SYSREAD, 222
OPERATING-SYSTEM, 224
PHP_CLI, 231
PHP_PATH, 231
POPUP_WINDOW_HEIGHT, 221
POPUP_WINDOW_WIDTH, 221
PROXY_FTP, 227
PROXY_HTTP, 227
REG_DEMO, 232
REG_KEY, 232
REG_NAME, 232
REPARSE_ACTION_CHECKED, 232
REPARSE_ACTION_FIXED, 232
REPARSE_SOURCE_CHECKED, 232
REPARSE_SOURCE_FIXED, 232
SAFE_TEMPLATE, 234
SAVEPAGE-PASSWORD, 233
SCHEDULE, 233
SECRET_FREQUENCY, 223
SECURE_UMASK, 224
SERIALIZER, 232
SITE-DIR, 224
SORT_ROLES, 235
SSH-CLIENT, 222
SSH-FILED, 222
SSH-OPTIONS, 222
SYSTEM-LANGUAGE, 225
SYSTEM_LOCALE, 225
TEMPLATE_CHAIN, 234
TEMPLATE_EWE, 234
TOPTABLE, 221
transfer, 234
UMLIGHT_PLUGIN, 235
USER_LOCALE, 225
Konfigurationsdatei
Suche, 200
Suchindex, 182
Konfigurationsdateien, 28
anzeigen lassen, 28
bearbeiten, 28
Bearbeitungsansicht, 28
editierbar machen, 29
Editor, 28
Übersicht, 28
Kontext beim Suchergebnis anzeigen lassen, 205
Konventionen, 1
Kopie erstellen-PlugIn, 103
Kopien von Dokumenten erzeugen, 103
kopieren
Systemdateien, 60

L

Lastverteilung, 124
LDAP, 77, 79, 84
ADIR, 84
Anmeldung, 78
Attribute, 80
Benutzer-Account, 78
Benutzeranbindung, 83
Einschränkungen, 78
imperia -Account, 78
imperia als Frontend, 83
Konfiguration, 79
PAM, 84
Passwort-Kodierung, 81
Passwort-Verschlüsselung, 81
Rollen-Attribute, 82
Rollen-Container, 81
Schreibzugriff, 83
Superuser-Rolle, 82
Superuser-Zugang, 82
User-Container, 80
LDAP-Filter, 81
Rollen-Identifikation, 82

LDAP-Perl-Modul, 77
LDAP-Zugang
  Informationen, 79
ldap.conf, 79
li_content, 174
li_snippet, 174
Link, 126-127, 133
Link-Checker, 27
Link-Text, 126
Linkcheck, 133
Linkpüfung, 134
Links, 45, 48
Linktext, 127
Linktool, 157
Linux
  MySQL, 20
  Oracle, 19
  PostgreSQL, 18
Liste, 130
  RDBMS-Treiber, 18
Liste einschränken, 75, 77
Livesystem, 173
Livesysteme, 1
Lizenzierung
  erworbene Funktionen, 6
Lizenznehmer, 6
Log-Datei
  Hermes, 169
Login, 77
Login-Maske, 5
LongReadLen
  Dokumentenhaltung, 16
LongTruncOk
  Dokumentenhaltung, 16
Löscheinträge, 136
löschen
  Benutzer, 77
  Codeinclude, 57
  Codeinclude-Verzeichnis, 57
  Flexmodul, 51
  Freischalt-Trigger, 10
  Metadatei, 47
  Rolle, 75
  Rubrik, 45
  Slotmodul, 54
  Systemdienst, 13
  Tabellen, 17, 21
  Template, 49
  Untermenü, 129
  Verbindung, 94
  Workflow, 90, 95
  Workflow-Schritt, 93
Löschen von Rubriken im Media Asset Management, 142
lösen
  Verbindung, 94

M

Mailsystem
  Internes, 39
Maintenance Mode, 5
MaintenanceNoLogin, 5
MAM, 157
  Flash-Mehrfachupload, 154
mam_spool_import.pl, 143
mask, 170
match, 199
MAX_PAGE_LIST, 209
MAX_PAGES, 209
maxDepth, 197
maxSuggestions, 216
Media Asset Management, 132
  Allgemeines, 140
  Allgemeines zu Asset-Rubriken, 140
  Asset-Rubrik anlegen, 140
  Asset-Rubrik bearbeiten, 142
  Asset-Rubrik löschen, 142
  globale Suchfilter, 146
  Konfiguration, 146
  Template, 153
  Zip-Import, 142
Mehr-Augen-Prinzip, 110
Mehrsprachigkeit, 119
memberuid, 82
memuseControl, 197
Menü
  bearbeiten, 128
  Menüpunkt hinzufügen, 128
  Zugriff, 128
  Zugriffsrechte, 128
Menü bearbeiten, 128
Menüabkürzungen, 126
Menüpunkt hinzufügen
  Menü, 128
  Untermenü, 128
Menüstruktur, 129
Meta-Import, 172
Meta-Information Bearbeiten
  Rubriken-Informationen, 70
Meta-Tags indizieren, 191
Metadatei
  anlegen, 45
  bearbeiten, 47
  bestimmen, 71
  hochladen, 46
  löschen, 47
Metadateien, 45, 170
  Rechte, 45
Metadateien und Templates auswählen
  Rubriken-Eigenschaften, 69
Metadaten setzten, 118
Metadatenselektion, 118
Metaedit, 116
Metafeld
  Template, 132
  Trigger, 9
Metafelder als Ausschlusskriterium, 190
Metafelder indizieren, 198
Metafelder nicht indizieren, 190
Metagruppen dynamisch benennen, 199
Metainformationen transformieren, 102
Metainformationen überprüfen, 102

metaKey, 190
Metakommentare indizieren, 186
minChars, 216
minValidCharacters, 206
Model, 130
MS-SQL, 14
Mülleimer, 119
multivalued, 82
MVC, 130
MySQL, 14
  Linux, 20
  Windows, 20

N

Name, 50, 52, 77
  Rubriken-Informationen, 70
Namespace
  OCE, 22
navigation_generator.conf, 173
navigation_generator.conf.sample, 173
Navigationselemente, 173
Navigationserzeugung, 173-174
  <!--XX-active-->, 174
  <!--XX-current-->, 174
  <!--XX-description-->, 174
  <!--XX-directory-->, 174
  <!--XX-filename-->, 174
  <!--XX-level-->, 174
  <!--XX-levelindex-->, 174
  <!--XX-link-->, 174
  <!--XX-linktarget-->, 174
  <!--XX-name-->, 174
Neue Hauptrubrik, 40
NodeID, 171
Notification, 2
Notification File, 13

O

Objektklassen
  imperia -Benutzer, 83
  Rolle, 83
Obligatorische Angaben, 74, 76
OCE, 21
  Bedingung, 23
  Hide, 22
  Hilfe, 23
  Icon, 23
  Label, 23
  sichtbar, 23
  Skript hinzufügen, 23
  Tooltip, 23
  URL, 23
  Zugriffsrecht(e), 23
OCE-Skript
  hinzufügen, 23
OCE-Skripte
  imperia , 24
oce_clone, 24
  __imperia_children, 25
  __imperia_created, 24
  __imperia_meta_expand, 24
  __imperia_modified, 24
  __imperia_node_id, 24
  __imperia_parent, 25
  __imperia_uid, 24
oce_delete, 25
  __imperia_node_id, 25
  copies, 25
  del_develop, 25
  directory, 25
  filename, 25
oce_edit, 24
  __imperia_node_id, 24
  __imperia_quick_edit, 24
  finished, 24
oce_erase, 25
  __imperia_node_id, 25
  force, 25
oce_history, 25
  __imperia_node_id, 25
oce_info, 25
  __imperia_node_id, 26
oce_linkcheck, 26
  __imperia_node_id, 26
  __imperia_quick_edit, 26
  finished, 26
oce_reparse, 26
  __imperia_node_id, 26
  finished, 26
  publish, 26
oce_unpublish, 26
  __imperia_node_id, 26
  directory, 26
  filename, 26
ODBC_MSSQL, 14, 17
One-Click-Edit, 21
One-Click-Edit-Konfiguration, 22
Onlinehilfe, 61
Operation, 44
Optionale Angaben, 74, 76
Optionen
  Hermes, 167
Optionen bei der Indexerstellung, 187
Optionen in Suchanfragen, 201
Options-Direktiven für Suchanfragen, 204
Oracle, 14, 17
  Linux, 19
Ordner
  Untermenü hinzufügen, 127

P

PAM, 82, 84
Parameter, 44
parseroptions-Direktiven für Suchanfragen, 208
passiv
  FTP, 8
passives FTP
  Zielsystem, 8
Passwort
  Dokumentenhaltung, 15
  gültig, 76
  wiederholen, 76

Zielsystem, 8
Passwort-Kodierung, 81
Passwort-Verschlüsselung, 81
PDFINFO, 186
PDFTOTEXT, 186
PER_PAGE, 210
percentPrecision, 205
perl, 50, 52
Perl-Modul
LDAP, 77
Pfad zum Excel-Parser der Suche, 186
Pfad zum PDF-Parser der Suche 1, 186
Pfad zum PDF-Parser der Suche 2, 186
Pfad zum Powerpoint-Parser der Suche, 187
Pfad zum Word-Parser der Suche, 186
Pfade von Suchergebnissen gemappt ausgeben, 202
phpEnable, 205
phpPath, 205
Platzhalter, 119
Plug-In, 170
Plug-In-Tabelle, 169
PlugIn
interaktiv, 96
nicht interaktiv, 96
PlugIn-Leiste, 91
PostgreSQL, 14, 17
Linux, 18
SuSE 7.1, 19
Windows, 19
Prefilter, 158
process_ssi, 209
Protokolle, 7
Publikationssdatum, 120
publish
oce_reparse, 26
Publizieren nach Workflow, 99
Publizieren und Löschen, 99

Q

queryMode, 206

R

RDBMS
spezifische Treiber, 18
RDBMS-Treiber
Aufstellung, 18
RDN-Attribut
Benutzer-Eintrag, 83
Rolle, 83
Rechte, 45, 48, 61, 89
Datei, 60
Metadateien, 45
Rubrik, 43
Workflow, 89
Rechte entziehen (-), 44, 61
Rechte hinzufügen (+, 61
Rechte hinzufügen (+), 44
Rechte setzen (=), 44, 61
Rechtevergabe, 60
Referenz, 132
Referenzen, 133, 157
darstellen, 134
parsen, 133
Referenztabelle, 133
Registrierungsschlüssel, 6
regkey.bin, 6
rekursive Indizierung aktivieren, 197
reparsen
Template, 58
Reparser, 58
resultFilterPlugin, 207
Rolle, 1, 126
anlegen, 73
bearbeiten, 75
löschen, 75
Rollen-ID, 75
Rollen, 44, 73, 75, 127
Rollen-ID, 75
Rollendaten
bearbeiten, 75
Rolleneinstellungen, 75
Rollenzugehörigkeit, 74, 76
Rubrik
anlegen, 40
bearbeiten, 41
löschen, 45
Rechte, 43
Reihenfolge ändern, 44
Zugriff, 71
Rubrik-ID, 171
Rubriken, 40, 69
Zugriffrechte, 69
Rubriken-ACL-Verwaltung, 43
Rubriken-Eigenschaften
Asset-Rubriken, 69
bestimmen, 69
Metadateien und Templates auswählen, 69
Rubriken-Information, 69
Workflow-Auswahl, 69
Zugriffsrechte, 69
Rubriken-Information
Rubriken-Eigenschaften, 69
Rubriken-Informationen
Beschreibung, 70
Dateiname, 70
eingeben, 70
Meta-Information Bearbeiten, 70
Name, 70
Verzeichnis, 70
Rubriken-Metainformationen, 41
Rubriken-Metainformationen bearbeiten
bearbeiten, 41

S

Schredder, 120
Script, 218
Server
Freischalt-Trigger, 9
SFTP-Transfer, 7
showContext, 205
SID

Dokumentenhaltung, 16
Site-Verzeichnis, 8
site/flex, 51
site/live/N, 135
site/meta, 135
site/Slot, 53
site_hermes, 166
site_hermes.pl, 166
SiteActive, 2
SiteActive-Basisverzeichnis, 12
SiteActive-Erstellung, 12
Sitemap, 173
Slotmodul, 52
bearbeiten, 53
erzeugen, 52
hochladen, 53
löschen, 54
Slotmodul-Verwaltung, 52
SmartMeta, 121
sortBy, 210
Sortierung, 158
Speichergröße für Indexerstellung, 197
Speicherort des Index für Suchanfragen angeben, 204
Speicherort des Suchindex bestimmen, 196
spezifische Treiber
RDBMS, 18
Sprache
Standard, 76
SQL-Lösch-Statements
anzeigen, 17
SQL-Server
Windows, 21
SQL-Statements
anzeigen, 17
Standard-Konfiguration
Hermes, 170
Standardfarben ändern
Stylesheet, 36
Standardprodukt, 2
Statements
Dokumentenhaltung, 16
Steuerung der Ausgabe von Suchergebnissen, 202
STRICT_PATHS, 139
Struktur, 40
Flexmodule, 50
Metadateien, 45
Slotmodule, 52
Templates, 47
Stylesheet
Standardfarben ändern, 36
Such-Metagruppen definieren, 187
Suchdomänen einrichten, 183
Suchergebnis-Seite angeben, 203
Suchindex, 181
Suchindex erstellen, 212
Suchmuster für Metagruppen definieren, 199
Suchskripte erstellen, 213
Suchtemplates, 202
Suchtemplates mit PHP, 205
Suchtiefe beim Indizieren, 197
suggest, 216
SUGGESTIONS, 217
Symbol, 126-127
Symbolische Links bei der Indexerstellung, 197
Symbolliste, 127
Syntax der Volltextsuche, 181
System
Konfiguration, 28
Konfigurationsdateien, 28
System-Conf Variablen
GLOBAL_METAKEYS, 132
System-Sprache, 27
system.conf, 79
system_files.conf, 60
Systemdateien
abgleichen, 60
kopieren, 60
übertragen, 60
Systemdienst
anlegen, 11
Dateien ausschließen, 13
Intervall, 11
mehrere Alarm-Verzeichnisse, 12
mehrere Verzeichnisse überwachen, 12
Verzeichnisse ausschließen, 13
Systemdienste
ändern, 13
Anzeige sortieren, 13
Ausführungszeitpunkt, 12
Beschreibung, 12
Dienstnummer, 12
Intervall, 12
löschen, 13
SiteActive-Basisverzeichnis, 12
SiteActive-Erstellung, 12
SiteActive-Template, 12
SiteActive-Zielseite, 12
Speicherdatei, 11

T

Tabellen
erzeugen, 17, 21
löschen, 17, 21
Template, 96, 217
bearbeiten, 49
bestimmen, 71
hochladen, 48
löschen, 49
Mediendatenbank, 153
reparsen, 58
Template-Prozessor, 28
Template-Reparser, 58
Template-Vorschau, 49
Templates, 47
temporäres Arbeitsverzeichnis für Suchindex bestimmen, 196
Tooltip
OCE, 23
Transfer, 173
transfer_cat_id, 173

Transformation-PlugIn, 102
Treiber-Optionen
  Dokumentenhaltung, 16
Trigger, 8
  Benutzerbedingung, 9
  Freischalt-Trigger, 8
  Immer, 9
  Metafeld, 9
  Zielsystem, 9
Trockenlauf, 17
Typ, 50, 52
Typ "TEXT"
  Dokumentenhaltung, 16

U

Übersicht
  Konfigurationsdateien, 28
Übersichtsseiten, 2
übertragen
  Systemdateien, 60
Übertragungsprotokolle, 7
uid, 82
uidnumber, 82
ul_content, 174
ul_snippet, 174
Unter-Workflow, 121
Untermenü
  anlegen, 126
  löschen, 129
  Menüpunkt hinzufügen, 128
Untermenü hinzufügen
  Menü, 127
Unterrubrik
  anlegen, 40
upload
  Codeinclude, 55
URI
  Zielsystem, 8
URI-Konflikte auflösen, 105
URL
  OCE, 23
User-Container, 80

V

verbinden
  Workflow-Schritt, 94
Verbindung
  löschen, 94
  lösen, 94
verbosityLevel, 197
verfeinertes Template für Suchanfragen angeben, 203
verfügbare Rollen, 71
verschieben
  Workflow-Schritt, 93
Versionierung, 132
Vertikalnavigation, 173
Verwaltung
  Workflow, 89
Verzeichnis
  Rubriken-Informationen, 70
Verzeichnisdienst
  Anbindung, 78
Verzeichnisdienst(e), 77
Verzeichnisgruppen im Suchindex, 192
Verzeichnisnamen als Index-Kriterium, 188
Verzeichnisse indizieren, 188
View, 130
Volltextsuche
  Alternative Schreibweisen / Worttrennungen, 177
  Anzahl der Treffer pro Seite, 210
  Anzeige von Suchergebnissen, 178
  Ausgabe von Meldungen bei der Indexerstellung, 197
  Ausgabe von Suchergebnissen steuern, 202
  automatische Selectbox für Gruppenauswahl, 203
  Beispiel für Metagruppen, 199
  benötigte Perlmodule, 180
  benutzerdefinierte Sortieroptionen unterdrücken, 210
  Berechnung der Treffer-Relevanz und Boni, 179, 184
  Cache-Größe für Index bestimmen, 195
  Dateinamen als Index-Kriterium, 189
  datumsbezogener Relevanzbonus, 211
  Default-Sortiermethode der Trefferliste, 210
  Direktiven für Suchoptionen, 204, 208
  Dokumentverzeichnis bestimmen, 196
  Domänen, 181
  Domänen in Suchanfragen, 201
  dynamisch benannte Meta-Gruppen, 199
  Einbindung von Stopwortlisten in die Indexkonfiguration, 186
  Einbindung von Stopwortlisten in die Suchkonfiguration, 200
  Einführung, 176
  Ergebnis-Template angeben, 203
  Ergebnisfilter, 179, 207
  Erstellung eines Index, 212
  Gewichtung beeinflussen, 184
  Groß -und Kleinschreibung, 176
  Hervorhebung von Treffern einstellen, 206
  Highlighting in Suchergebnissen, 179
  Höchstzahl generierter Trefferseiten, 209
  Höchstzahl von Links auf Trefferseiten, 209
  Index-Speicherort für Suchanfragen angeben, 204
  Indexerstellung, 182
  indizierte Datentypen bestimmen, 183
  Indizierung aktivieren, 193
  Indizierungs-Konfiguration vererben, 194
  Kategoriserung von Suchanfragen, 177
  Kompressions-Plug-In auswählen, 195
  Konfiguration, 181
  Kontextanzeige bei Suchergebnissen, 179
  logische Operatoren, 178
  logische Verknüpfung von Metagruppen, 207
  maximale Suchtiefe, 197
  Meta-Angaben von PDF-Dateien indizieren, 191

Meta-Gruppen, 198
Meta-Gruppennamen festlegen, 198
Meta-Kommentare, 186
Meta-Tags indizieren, 191
Metadaten beim Indexieren ignorieren, 208
Metafelder als Ausschlusskriterium, 185, 190
Metafelder als Indexkriterium, 185
Metafelder als Kriterium zum Indizieren, 190
Metafelder indizieren, 198
Metafelder nicht indizieren, 190
Metagruppen, 187
Microsoft Office Dokumente indizieren, 180
Minimalanzahl feststehender Buchstaben, 206
nach Metainhalten filtern, 178
Nachkommastellen für PAGE_PERCENT, 205
Option-Blöcke, 187
Optionen in Suchanfragen, 201
PDF-Dokumente indizierenkripte für die Konfiguration, 180
Pfad zum Excel-Parser, 186
Pfad zum PDF-Parser 1, 186
Pfad zum PDF-Parser 2, 186
Pfad zum PHP-Interpreter angeben, 205
Pfad zum Powerpoint-Parser, 187
Pfad zum Word-Parser, 186
Pfade in der Ausgabe mappen, 202
phonetische Ähnlichkeit, 177
PHP im Suchtemplate parsen, 205
Phrasensuche, 177
Reguläre Ausdrücke, 182
rekursive Indizierung aktivieren, 197
Schnellstart, 180
Skripte für die Konfiguration, 179
Sonderzeichen und Interpunktion, 176
Speicher kontrollieren, 197
Speicherort des Index bestimmen, 196
Speicherort für Datei-Cache bestimmen, 195
Sprachspezifische Synonyme definieren, 201
Standard Such-Template bestimmen, 203
Startseite der Ergebnisliste bestimmen, 209
Steuerung der Kontextanzeige, 205
Stopwörter, 177
Suchbegriffe gruppieren, 178
Suchdomains konfigurieren, 183
Suche nach Metagruppen ohne Suchwort, 206
Suchkonfiguration, 200
Suchmuster für Metagruppen definieren, 199
Suchskripte erstellen, 213
Symbolische Links, 197
Syntax, 181
Syntax für Suchbegriffe, 176
Templates, 202
temporäres Arbeitsverzeichnis, 196
Trefferliste sortieren, 210
Umlaute, 177
URL-Suche, 177
verfeinertes Such-Template bestimmen, 203
Verwendung mehrerer Wildcards ermöglichen, 206
Verzeichnisnamen als Index-Kriterium, 188
Verzeichnisse gruppieren, 192
Verzeichnisse indizieren, 188
Voraussetzungen, 180
Wildcards, 176
Wortstammsuche, 178
Zeichensatz für Verzeichnisse festlegen, 194
zusätzliche Variablen, 206
Vorname, 77
Vorschau
Template, 49
Vorschlag, 217
Vorschläge, 215

W

Wartung, 5
Wartungsarbeiten, 5
Wartungsmodus, 5
mit Superuser-Zugriff, 6
WebDAV, 7
Wiedervorlagedatum, 124
Windows
MySQL, 20
PostgreSQL, 19
SQL-Server, 21
Windows®-Systemdienst
Hermes, 166
Workflow, 1, 132-134
anlegen, 89
bearbeiten, 90, 95
festlegen, 70
löschen, 90, 95
Rechte, 89
Zugriff, 89
Workflow einbinden-PlugIn
Workflow einbinden, 121
Workflow Plug-In
ABC1, 133
ABC2, 134
ABC3, 134
Workflow-Auswahl
Rubriken-Eigenschaften, 69
Workflow-Einstiegspunkt, 70
Workflow-PlugIn
Annotate, 98
Archiv-Bereiniger, 98
Asset hochladen, 122
Asset-Aktionen, 96
Asset-Variante, 103
Bearbeiten, 99
Bearbeiten +, 101
Bedingung, 111
Definition, 91
Depublikationsdatum, 110
Depublizierung, 108
Dokumentenauswahl, 105
Eskalation-Ende, 109
Eskalation-Start, 108
Existenzprüfung, 109
Genehmigung, 110
Genehmigung +, 112
HTML-Konverter, 102
Inhaltsprüfung, 102

Kopie erstellen, 103
Lastverteilung, 124
Mehrsprachigkeit, 119
Metadaten setzten, 118
Metadatenselektion, 118
Metaedit, 116
Mülleimer, 119
Platzhalter, 119
Publikationssdatum, 120
Publizieren nach Workflow, 99
Publizieren und Löschen, 99
Schredder, 120
SmartMeta, 121
Transformation, 102
Wiedervorlagedatum, 124
Workflow-PlugIns , 96
Workflow-Schritt
anlegen, 92
Definition, 91
duplizieren, 93
Eigenschaften, 92
löschen, 93
verbinden, 94
verschieben, 93
Zugriffsrechte, 94
Workflow-Schritte
Änderungen, 88
Aufbau, 88
Workflows, 88
Aufbau, 88
verwalten, 89
WYSIWYG-Editor, 62

X

XLS2CSV, 186
XML, 170
XML-Datei, 173
XML-Import, 172

Z

Zeichensatz für Verzeichnisse festlegen (Suche), 194
Zieldatei
Freischalt-Trigger, 8
Zielformat
Freischalt-Trigger, 8
Zielrubrik
Hermes-Import, 171
Zielsystem, 1, 132-133, 135
Anmeldekennung, 8
IP-Adresse, 8
passives FTP, 8
Passwort, 8
Trigger, 9
URI, 8
wiederherstellen, 137
zusätzlich, 8
Zip-Datei Import, 142
Zugriff
Menü, 128
Rubrik, 71
Workflow, 89
Zugriffsrecht
Bearbeiten, 94
entfernen, 95
Lesen, 94
Löschen, 94
Zugriffsrecht(e)
aberkennen, 72
OCE, 23
zuweisen, 71
Zugriffsrechte, 74, 76
einstellen, 94
Menü, 128
Rubriken, 69
Rubriken-Eigenschaften, 69
Workflow-Schritt, 94
zusätzlich
Zielsystem, 8
Zusätzliche Links, 126
zuweisen
Zugriffsrecht(e), 71